Tampere University

Oskari Burri

PROVIDING MACHINE LEVEL DATA FOR CLOUD BASED ANALYTICS

Faculty of Engineering and Natural Sciences
Master of Science Thesis
April 2019

ABSTRACT

OSKARI BURRI: Providing machine level data for cloud based analytics
Tampere University of Technology
Master of Science Thesis, 65 pages, 8 Appendix pages
Automation engineering
Major: Informatics in Automation
Examiners: Professor Matti Vilkko and Assistant Professor David Hästbacka

Keywords: cloud, industry 4.0, analytics, edge computing

The target of this thesis was to investigate the problems, possibilities and background of transporting data collected in a manufacturing process to a cloud environment for analytics. The underlying reason for this was to spot themes and topics that need to be addressed when planning a real-life project aiming to improve manufacturing with cloud manufacturing.

To achieve this, first the underlying theories were studied to understand why we actually need cloud analytics and what kind of paradigms could help in achieving it. The reasoning was mostly based on themes revolving around Germany's Industry 4.0 initiative. After that, the state of the art was researched to understand what has been already done and how some fundamental problems have been solved. From this base, requirements for a solution model were made. After requirements for a good system were formulated, some of the most promising technologies were researched and after that evaluated according to the requirements.

This thesis was able to highlight many core problems that modern cloud analytic systems for manufacturing will face. The most central technological findings revolve around a vital need for industry wide standardization, which is a theme that requires both customers and vendors: customers need to demand vendors to embrace common open standards and vendors need to respond to this need.

Another core problem discussed in this thesis was the usage of different cloud services. Many manufacturing giants offer their own platforms for providing cloud analytics, but also bring some fundamental problems to play. On the other hand open cloud platform providers like Microsoft Azure and Amazon Web Services offer a very large offering of components, services and applications with competitive pricing.

Altogether this thesis produced some topics that require a thorough analysis when planning a system like this and some insight into the core problems. In addition, some recommendations were produced on how to tackle problems in ways that transition well in to the future as the field is evolving rapidly and future compatibility is important.

TIIVISTELMÄ

OSKARI BURRI: Tampereen teknillisen yliopiston opinnäytepohja
Tampereen teknillinen yliopisto
Diplomityö, 65 sivua, 8 liitesivua
Automaatiotekniikan diplomi-insinöörin tutkinto-ohjelma
Pääaine: Automaation tietotekniikka
Tarkastajat: professori Matti Vilkko ja Assistant Professor David Hästbacka

Avainsanat: cloud, industry 4.0, analytics, edge computing

Tämän diplomityön tarkoituksena oli tutkia datan keräystä tuotantoprosessista pilviympäristöön analytiikkaa varten, sekä siihen liittyviä ilmiöitä, ongelmia sekä mahdollisuuksia. Motivaatioina tähän kaikkeen oli havaita mahdollisia aiheita ja aihepiirejä, joita tulisi tutkia ja arvioida reaalimaailman projektia suunnitellessa.

Tavoitteisiin pääsemiseksi aluksi tutkittiin teorioita, joiden avulla haettiin ymmärrystä miksi oikeastaan pilvianalytiikkaa tarvitaan ja minkälaiset ratkaisumallit voisivat auttaa sen toteuttamisessa. Käsitellyt teemat liittyivät hyvin vahvasti Saksan Industry 4.0 hankkeeseen ja siihen liittyvään tutkimustyöhön. Tämän jälkeen tarkasteltiin nykytilaa ja minkälaisilla ratkaisuilla ongelmaa yleensä on ratkaistu. Tämän pohjalta muotoiltiin vaatimuksia ratkaisumallille. Vaatimusmäärittelyiden pohjalta alettiin tutkia lupaavimpia teknologioita ja tekniikoita vaatimusten pohjalta.

Työ onnistui nostamaan esille useita keskeisiä ongelmia aiheuttavia teemoja, mitä modernit pilvianalytiikkaratkaisut joutuvat kohtaamaan. Keskeisimmät tekniikkaan liittyvät löydökset ovat vahvasti kytköksissä toimialojen laajuisen standardoinnin tarpeeseen. Tämä on teema, joka vaatii toimenpiteitä niin asiakkailta kuin toimittajilta: asiakkaiden tulee vaatia toimittajia perustamaan ratkaisunsa avoimille ja yhteisille standardeille ja toimittajien tulee vastata tähän tarpeeseen.

Toinen keskeinen käsitelty ongelma oli erilaisten pilvipalveluiden vertailu ja niistä oikean valinta. Useat valmistavan teollisuuden toimittajajätit tarjoavat omia alustoja pilvianalytiikalle, jotka tarjoavat valmiita ratkaisuja ainakin joihinkin ongelmiin. Toisaalta avoimet pilvitoimittajat kuten Microsoft (Azure) ja Amazon (AWS) tarjoavat erittäin laajalla skaalalla erilaisia komponentteja, palveluita ja sovelluksia kilpailukykyisellä hinnoittelulla.

Yhteenvetona tämä diplomityö nosti ja tuotti pohdintaa useasta eri aihealueesta, mitkä vaativat huolellista analyysiä reaalimaailman ratkaisua suunnitellessa. Lisäksi pohdinnan perusteella pystyttiin muotoilemaan joitakin ratkaisuehdotuksia ongelmien välttämiseksi tavalla, jotka tulevaisuudessa tukevat käsitellyn alan nopeasti kehittyvää luonnetta ja korkeita vaatimuksia yhteensopivuudelle.

PREFACE

This thesis was commissioned by a customer of Solita Oy in order to acquire an academic view to their internal development plans and road maps. I would like to thank them for providing me with, although a demanding but a very interesting topic and excellent guidance throughout the process.

On Solita's side I would thank everyone who helped me through my thinking process, specially my supervisor Matti Partanen. It has been an immense advantage to have the resources and knowledge of the company's experts available for consultation during the writing process.

Finally I would like to thank my examiners Professor Matti Vilkko and Assistant Professor David Hästbacka for constructive feedback and support.

In Tampere, Finland, on 18 April 2019

Oskari Burri

CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

API	Application Programming Interface
AWS	Amazon Web Services
AMQP	Advanced Message Queuing Protocol
CPS	Cyber Physical System
CPPS	Cyber Physical Production System
DES	Discrete Event Simulation
ERP	Enterprise Resource Planning
ETL	Extract, transform, load
HTTP	Hyper Text Transport Protocol
IaaS	Infrastructure as a Service
IMS	Intelligent Manufacturing System
IoT	Internet of Things
IIoT	Industrial Internet of Things
IP	Internet Protocol
JSON	JavaScript Object Notation
MES	Manufacturing Execution System
MPP	Massively Parallel Processing
MQTT	Message Queuing Telemetry Transport
NIST	National Institute of Standards and Technology
OPC UA	Open Platform Communications Unified Architecture
PaaS	Platform as a Service
PLC	Programmable Logic Controller
REST	REpresentational State Transfer
RMS	Reconfigurable Manufacturing System
Saas	Software as a Service
SCADA	Supervisory Control And Data Acquisition
SSL	Secure Sockets Layer
SDK	Software Development Kit
TCP	Transmission Control Protocol
TLS	Transport Layer Security
WS EMS	WebSocket-based Edge MicroServer

1. INTRODUCTION

Collecting and utilizing data is one of the major trends of this decade. Already in 2013 IBM's CEO Ginni Rometty stated that "Data will be the basis of competitive advantage for any organization you run", and this has become rapidly a central ground truth when trying to stay competitive in the global market [1].

Gathering data in order to monitor and improve production has been around for decades. In factories this is usually done by local systems gathering data from the process control automation. The gathering and analyzing of the data is often the responsibility of a manufacturing execution system (MES), which resides on the third layer of the ISA95-model. MES applications are usually executed in an isolated factory network, which have no Internet access due security reasons. These systems access the data in the programmable logic either with a manufacturer specific, proprietary protocol or with an open protocol like OPC or lately OPC-UA.

One specially popular use case for data-analysis has been of predictive maintenance. It uses historical data to optimize service intervals and has provided massive savings through shorter machinery downtimes without excessive amount of service calls. In the recent years data analytics has adopted cloud platforms very rapidly, mainly due the flexibility, ease-of-implementation and high efficiency it provides [2].

The future of manufacturing is far from certain, but one recent development is widely supported as the most probable evolution path: Industry 4.0. It depicts a lot of different aspects of the manufacturing industry and one of it's core topic is networking. Specially the concept of "smart factories" rely on highly networked actors which can pass information to each other freely. [2]

Another key theme of Industry 4.0 is decentralization. This has for example a disrupting effect on thinking based on ISA95: information and data cannot be confined inside each layer. Instead it has to be accessible as easily as possible to maximize its value. This leads to a mesh architecture, shown later in Figure 2.5 [3]. It is clear that the current status quo in manufacturing needs to change towards a more networked and better integrated solution. [4]

This thesis was ordered by a Finnish food manufacturing company in order to understand better the process of gathering data for advanced analytics. Specially interesting for them are the different options available and understanding what are the critical questions you should ask when planning the implementation of the future analytics capabilities in their factories. The goals for them are quite common ones and have many similarities with the

goals presented in academia under the topics of IIoT and Industry 4.0: better visibility to the factory floor and a deeper understanding of the process [5, s 3-4]. However the current situation at customer is far from these goals - very little comparison is done between the consumed raw materials and the planned material consumption, information is transferred between systems by humans and recipes used are customized by operators and tracked nowhere. Addressing these issues is not particularly interesting, as endless companies have overcome them, often with a help of some of systems on the ISA-95 level 3. This thesis tries to help in looking further than that obvious next step, and to ensure that the decisions made support the following development towards capabilities depicted in Industry 4.0 as much as possible.

1.1 Research questions

This thesis seeks answers to the following questions: ”How should production data be gathered from an Siemens S7 process logic controllers?”, ”How should the gathered data be transferred to a cloud service?” and ”What kind of a cloud service should it be transported to for analytics?”. These questions were formed with the help of interviews documented in Appendix B and Appendix C they were researched in the context of the customer’s case to keep the scope feasible. Specially for the last question the number of researched platforms and softwares has to be limited due to the nature of the domain.

1.2 Methods

First the case and its scope is established with the help of customer interviews and literature. After establishing the problem and the factors affecting the possible solution, we identify the steps needed to solve the case. Then with the help of experts and literature we try to identify the relevant problems and decisions that need to be solved and made to implement an advanced analytics solution successfully.

This thesis does not seek to solve all arising questions, but to provide understanding of the domain and the inevitable problems and decisions that a project aiming to build an advanced analytics solution will face. The goal is to educate the reader to the subject and to give insights for the procurement of an cloud based production data analytics system.

1.3 Definition of analytics and advanced analytics

Throughout this thesis, the words ”analytics”, ”traditional analytics” and ”advanced analytics” are used. The first word is very hard to define, as it has became a catch-all term for various applications. Also people working in different fields and positions use the word very differently. In the scope of this thesis, ”analytics” covers all applications which aim to refine data into information and possibly to knowledge. [6]

”Advanced analytics” is defined by Gartner’s IT Glossary as ”autonomous or semi-autonomous examination of data or content using sophisticated techniques and tools, typically beyond

those of traditional business intelligence (BI), to discover deeper insights, make predictions, or generate recommendations. Advanced analytic techniques include those such as data/text mining, machine learning, pattern matching, forecasting, visualization, semantic analysis, sentiment analysis, network and cluster analysis, multivariate statistics, graph analysis, simulation, complex event processing, neural networks." [7]

These two definitions give us a quite good understanding of these two terms, but "traditional analytics" falls somewhere in the middle of these without any real definition. To find a definition for it, we use Frost and Sullivan's definition of the three generations of analytics. "Analytics 1.0" is based on Data Warehousing and Business Intelligence platforms rooted in the 1990s. This then evolved to "Analytics 2.0", which include newer platforms like Apache Hadoop to combat the data explosion created in the Internet era. Finally comes the current era, "Analytics 3.0" where we have more sophisticated data handling and insights and we can predict more than ever before. In this thesis the generation of "Analytics 1.0" is considered "traditional analytics", where as the following two generations are considered "advanced analytics" [8, p. 14]

2. UNDERLYING THEORIES

In this chapter the underlying theories and models considering this domain are discussed. We start with older and more established topics and proceed then to more modern and still rapidly evolving topics.

2.1 ISA-95 (IEC 62264)

ISA-95 is a widely accepted industry standard developed and maintained by The International Society of Automation (ISA), which is a technical society for industrial automation and instrumentation.

The need for ISA-95 originally rose from the need to integrate Enterprise Resource Planning (ERP) systems with operational production systems. It consists of models and terminology about i) information exchange between ERP systems and manufacturing operation systems ii) activities in manufacturing operations systems and iii) exchanged information within manufacturing operations systems. [9]

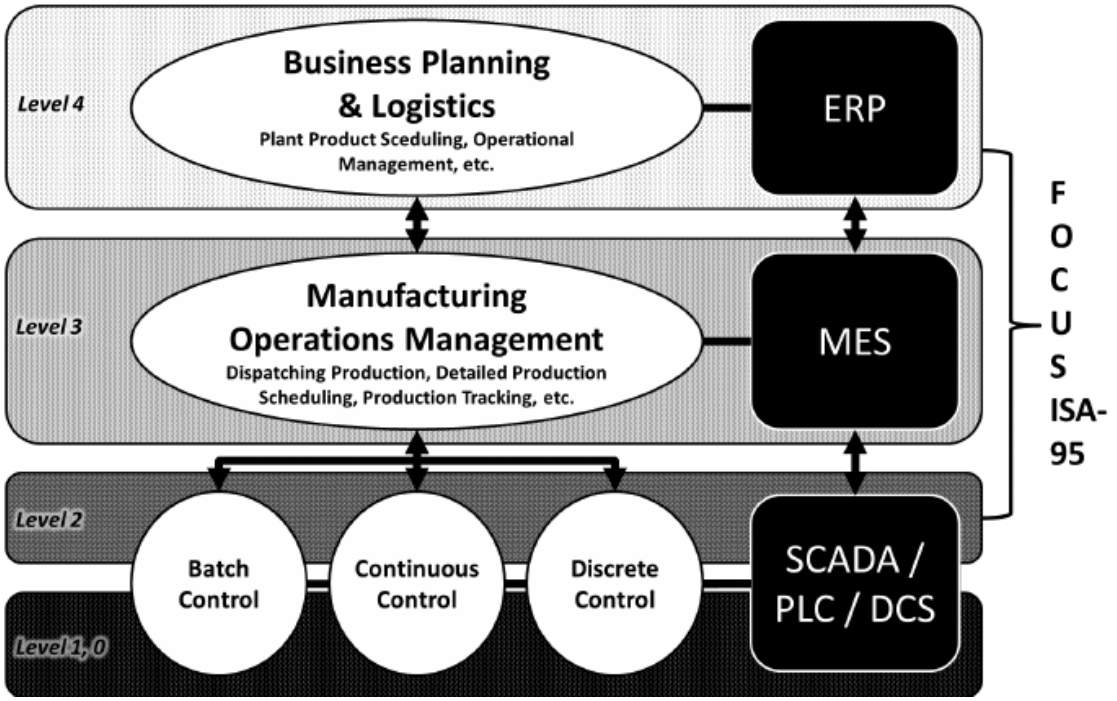

Figure 2.1. *Three tier architecture depicted in the ISA-95 standard [9].*

One of the standard's key models is the 4 or 5 level model shown in Figure 2.1 which is divided to 3 tiers. On the top at level 4 of the model resides the corporate management that usually uses an ERP software. On the level 3 is the MOM-layer which a MES system

is part of. On the bottom tier of the model are the levels 2, 1 and 0. On this tier reside the physical production (level 0), sensing and actuation (level 1) and on the level 2 the automation process control and supervision like Programmable Logic Controllers (PLC), Supervisory Control And Data Acquisition systems (SCADA) and Distributed Control Systems (DCS). [9]

The main focus of ISA-95 is to define and model the relations and communications between the systems on levels 4, 3 and 2.[[9]] However it is good to note that the ISA 95 is very principle-based and therefore more suitable when planning large systems with a certain degree of standardization. In daily life applying it's theoretical principles may be a bit of a burden. Instead it should be used as a design guide when designing basic structures for factory information architecture. [10]

2.2 Manufacturing Execution Systems

Gartner's glossary gives us the following definition: "Manufacturing execution systems (MES) manage, monitor and synchronize the execution of real-time, physical processes involved in transforming raw materials into intermediate and/or finished goods. They coordinate this execution of work orders with production scheduling and enterprise-level systems. MES applications also provide feedback on process performance, and support component- and material-level traceability, genealogy, and integration with process history, where required."

This definition can be roughly split into three fields:

- Managing, monitoring and synchronizing production
- Coordinating the execution with production scheduling and enterprise-level systems
- Providing feedback and history data

This means that in addition to running the production, MES's key responsibilities include a tight integration to the ERP. Before MES systems, this integration between the shop floor and management was mostly done manually, and had a resolution of days. [10] This tight integration also enables organizational vertical integration between management and production. This integration is of paramount importance for companies seeking to enjoy a competitive future. [10]

2.3 Analytics in Manufacturing

Frost and Sullivan identifies 4 key areas for manufacturing environments, where analytics is used. The first and most adopted area is "Process Improvements", which includes asset performance management, asset reliability, and production and supply-chain optimization. The second area with a much smaller adaptation is Resource Optimization , which mainly includes workforce management. The final two areas, which also have the smallest adoption

rates, are ”Risk and Security Portfolio Management” and ”Customer/Consumer Behavior”. [11]

The first topic could be summarized also as improving efficiency. Usually this is done by improving productivity without reducing quality. In [12] needs for analytics in manufacturing is categorized into 5 fields

- Reducing test time and calibration
- Improve quality (less scrap)
- Reducing warranty cost
- Improving yield
- Performing predictive maintenance

Traditional quality improvement programs also use statistics to some extent, but usually they are applied to relatively small sets of data. However recent advances in data processing have provided several new methods for analyzing large amounts of data, and these methods are usually referred to as big data analysis. [12]

Big data analysis, as its name implies, revolves around processing big amounts of data, usually ranging up from terabytes. It has an ability to find complex multivariate nonlinear relationships, run machine learning algorithms and even differentiate causation from correlation. Using big data methods has grown more widespread than ever before, and due the growth of cloud computing, also smaller companies in less data intensive business fields are able to start utilizing them. [5, 8, 12]

Currently manufacturing experts often rely on their domain knowledge to detect most important factors affecting efficiency, and then run experiments to validate their suspicions. This combined with the possibilities and tools provided by big data analytics provide a quick detection of root causes for problems and failures. [12]

In addition to more and better tools, there is an ongoing paradigm change in manufacturing automation. In the traditional approach data is analyzed retrospectively and problems are addressed after they have occurred. This analysis is also called descriptive analysis, as it determines what has happened. [5, 11]

The emerging trend is now towards predictive and/or proactive approach, where real-time data is analyzed with the methods of advanced analytics to predict the future and possible breakages associated with it. This development is driven by the costs associated with equipment downtime and decreased efficiency caused by problems not identified proactively. [11] In addition to predictive analysis, advanced analytics can also provide prescriptive analysis to provide suggestions how to act now to in order to achieve an optimal result. [5]

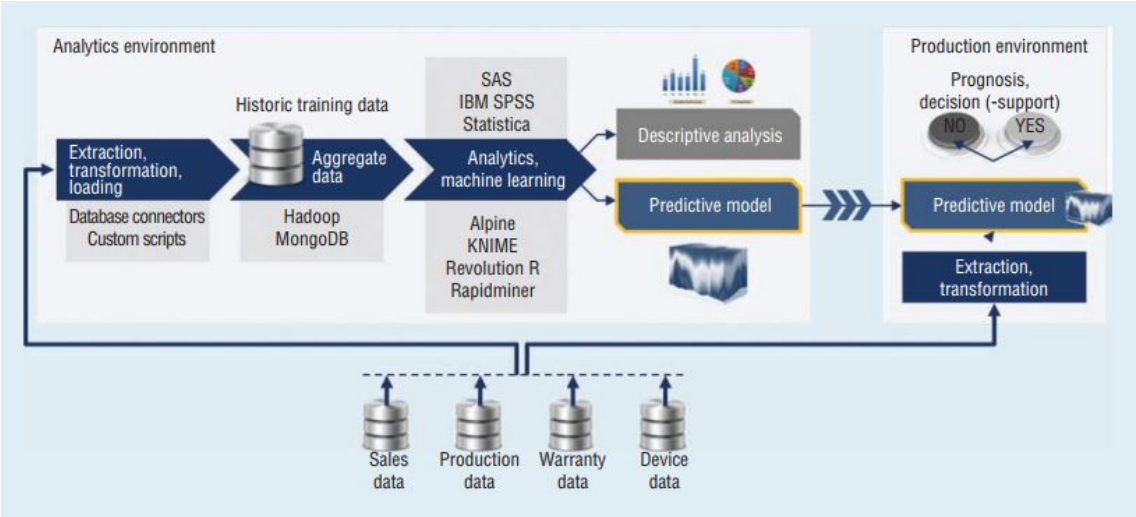

Figure 2.2. *An example high level architecture how an advanced analytics system could look like in manufacturing [12].*

Analytics can provide the feedback defined as one of the responsibilities of a MES system as described in section 2.2. However in the light of the following chapters a question arises regarding how tightly this kind of analytics should be tied to MES.

In Figure 2.2 is depicted an example data architecture of an manufacturing analytics solution, which produces predictive and descriptive analysis described previously. It uses historical data gathered from various sources to produce descriptive analysis and predictive models. The predictive models are then used for (real time) analysis to produce a prognosis from the future and give decision support.

2.4 Industrial Internet or Industrial Internet of Things

IIoT has many names, General Electric (GE) coined the term Industrial Internet, Cisco calls it the Internet of Everything and some Internet 4.0. Many industrial leaders forecast that it will bring unprecedented levels of growth and productivity over the next decade. [5]

The problem of IIoT has been the lack of clear vision what it is and what benefits it will provide. On consumer side, Internet has revolutionized how business is done, some business models have vanished and even more have emerged. But on the industry side it has been unclear, what direct benefits does connecting machine-to-machine networks to the Internet provide? To understand the answer to this question, you have to understand what IIoT is about. [5]

The main advantage of IIoT is undoubtedly increased visibility and insight into the companies operations and assets. The data gathered can be transformed with the help of analytics to a feedback loop for the business's processes. It may sound that these possibilities already exist in traditional M2M technologies, but the scale is much smaller compared to IIoT. In IIoT systems huge real time data streams can be analyzed in a cloud with advanced analytics

at wire speed providing reliable predictions about the future. Another advantage for IIoT is ”the power of 1 percent”. When speaking about large scale operations, like aviation, savings of one percent on fuel may result savings of 30 billion dollars per year. [5]

But what enables all this? Firstly the quality and intelligence of sensors has increased dramatically in the recent years, while simultaneously their size and price has gone down. These sensors are able to produce more and better data which can be fed to rapidly evolving advanced analytics method. Together with scalable and cheap computing power in cloud services, these analytics can provide better and more accurate insight and predict future problems with ever increasing accuracy. [5]

In addition to these enabling factors, a demand is also needed for things to change. In case of IIoT those are numerous: firstly the complexity of industrial systems are out-pacing human operators abilities to recognize and address problems or inefficiencies. Secondly the maturity of cloud systems and wireless networks have improved to such levels, that they can be considered as primary platforms for new projects. [5]

2.5 Edge Computing

Edge computing refers to doing computational operations on the edge of the network which [13] describes as ”any computing and network resources along the path between data sources and cloud data centers”. It is a very popular topic when discussing cloud solutions and it is at the top of the Gartner’s hype cycle for Cloud Computing in 2018 [14]. In this case investigating edge computing is relevant because the customer’s requirements include real-time capabilities.

2.5.1 Why do we need Edge Computing

In a global scale, can three major drivers be identified as drivers for edge computing adaptation:

- Push from cloud services
- Pull from IoT
- Change from data consumer to producer [13]

The first point has emerged as processing data in the cloud has proved to be better than doing it on the edge. This has lead to architectures where all the data processing is done on the cloud. However, due the increased amount of data produced by different devices, network capacity has started to become the limiting factor. Also for application with hard real-time requirements the delays of the communications may be unacceptable. [13]

The second reason is related to the first one: the amount of data produced on the edge is increasing with a rapid rate. Therefore it is not feasible to assume all the data will be sent to

a cloud service for processing, instead it will be consumed also at the edge of the network. Also many IoT-devices are energy constrained and therefore offloading some computation or communication to an edge device may be desirable. [13]

The last point is caused by the fact that traditional data consumers have started to produce more and more data themselves, for example consumers instead of watching some videos now more and more publish media themselves to for example social media [13]. In a factory setting this is even more true: when improving a manufacturing process, very often the data source and the data user are both located inside the factory.

For industrial usages the Industrial Internet Consortium's whitepaper adds also one additional aspect: compliance, data privacy and data security concerns [15]. With edge computing it is possible to reduce or eliminate totally sensitive data leaving the facility in question.

2.5.2 Benefits of Edge Computing

The most obvious benefits of processing data locally is the reduced latency [5]. This may enable doing control decisions based on the collected data within milliseconds. Also allowing a system to function independently increases the its resiliency against connectivity and availability problems. [15]

Modern systems also produce ever increasing amounts of data. Sending all of it to a cloud service may be possible, but it may require additional investments to connectivity and become also expensive due many cloud platform's dynamic pricing based on ingested data amount. Therefore edge computing may provide significant savings by analyzing bulk of the data on-site, and only sending aggregated results to the cloud. [15]

2.6 Industry 4.0

Industry 4.0 is a term originating from Germany, which describes a vision for the fourth industry revolution, which is based on Cyber-Physical Systems (CPS) like described in Figure 2.3.

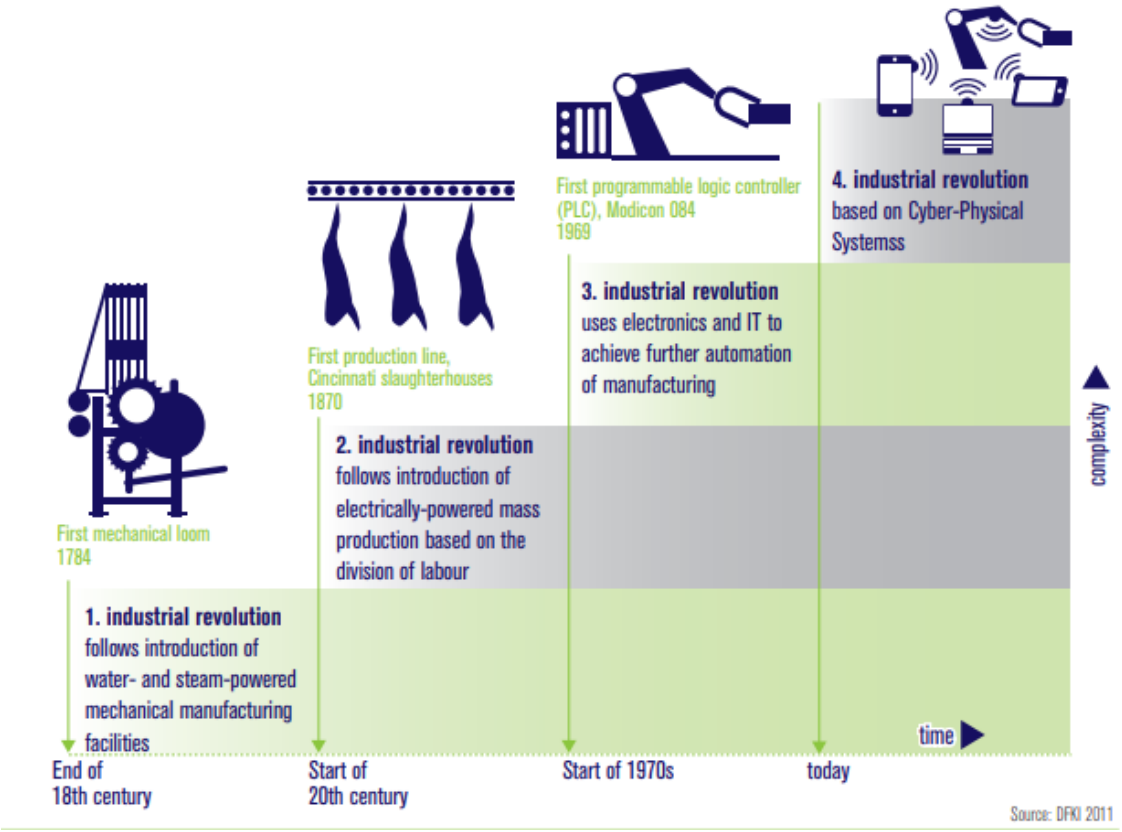

Figure 2.3. *The four industrial revolutions [16].*

The main targets for it are

- Meeting individual customer requirements
  - Making batch size 1 profitable
- Flexibility
  - CPS-based ad-hoc networking enabled dynamic configuration
- Optimized decision-taking
  - Increasing end-to-end transparency to allow better decisions to be made
- Resource productivity and efficiency
  - CPS allow better optimization of systems to increase productivity
- Creating value opportunities through new services
  - New innovative services can be created from big data recorded by intelligent devices
- Responding to demographic change in the workplace

  - Enable diverse and flexible career paths that allow people to work and remain productive for longer
- Work-Life-Balance
- A high-wage economy that is still competitive

[16]

To achieve these things, the Industry 4.0 Working Group believes that action is required on the 8 following key areas:

Standardization and reference architecture
Industry 4.0 involves networks spanning across multiple companies through value networks. This kind of partnership is only possible if a single set of standards are developed. Also a standard architecture is required to describe these standards and to facilitate in their implementation. [16]

Managing complex systems
Products and systems are becoming increasingly more complex. Planning and explanatory models can provide a basis for managing this growing complexity. Engineers should therefore be equipped with tools and methods to develop such models. [16]

A comprehensive broadband infrastructure for industry
Reliable and generally high-quality networks are a key requirement for Industry 4.0 for enabling extensive networking between different actors [16].

Safety and security
Both safety and security are essential to Smart Factories. Production and products should not pose a danger either to people or to the environment. On the other hand the data and information they contain should be protected from misuse and unauthorized access. [16]

Work organization and design
In smart factories the role of employees will change significantly. A socio-technical approach needs to be implemented to offer workers greater responsibility and enhance their personal development [16].

Training and continuing professional development
As the job and competence profile of workers is changed drastically by Industry 4.0, it is necessary to implement training strategies that fosters learning and enables lifelong learning [16].

Regulatory framework
Existing legislation needs to be adapted to take new innovations into account. The challenges faced include the protection of corporate data, liability issues, handling of personal data and trade restrictions. In addition to legislation, actions are also required on behalf of businesses like guidelines, model contracts, company agreements and self-regulation initiatives such as audits. [16]

Resource efficiency

Manufacturing industry consumes large amounts of raw materials and energy and poses a number of threats to the environment and security of supply. Industry 4.0 will increase efficiency and productivity, but requires vast amounts of investments to implement. It will be necessary to calculate the trade-offs between investments to smart factories and potential savings. [16]

These goals could be reached by implementing CPS and creating horizontal integrations through value networks, end-to-end digital integration of engineering across the entire value chain and finally with vertical integration and networked manufacturing systems. This thesis will mostly cover themes from the last topics. One key enabler for creating networks and vertical integration to entire manufacturing systems and factories is Internet of Things. [16]

2.6.1 Cyber physical systems

A cyber-physical system (CPS) is an entity composed from both cyber and physical parts. Usually a CPS consist of one or more microcontroller(s) which then interact somehow with the real world and processes the data obtained. In addition a CPS needs some kind of an networking interface to exchange data with other systems or a cloud. The ability to change data is the most important feature of a CPS. [17]

E. A. Lee and S. A. Seshia describe CPS with the following words: ”As an intellectual challenge, CPS is about the intersection, not the union, of the physical and the cyber” [18]. In R. Iqbal et. al. describes CPSs as ”state of the art cloud-based architectures” and point out, that the computation and communication in and to the cyber part of the CPS include several Big Data operations like sensing, storing and processing large amounts of heterogeneous data.

Like described in section 2.6, CPS are a key enabler for the futures manufacturing. Therefore it can be beneficial to examine the benefits and challenges provided by them when developing plans and infrastructure for a manufacturing environment. CPS in a manufacturing environment are called Cyber physical production systems or CPPSs. [2]

2.6.2 Cyber physical production systems

A cyber-physical production system (CPPS) is a combination of autonomous and cooperative elements. In [19] its main characteristics are described as intelligence, connectness and responsiveness. This means a CPPS should be capable of collecting information from its environment and information networks and based on that act autonomously and react to internal and external changes.

In Industry 4.0 CPSs are used to build a CPPS platform to connect virtual and physical world and to allow equipment in a smart factory to be more intelligent [2]. CPPS are often

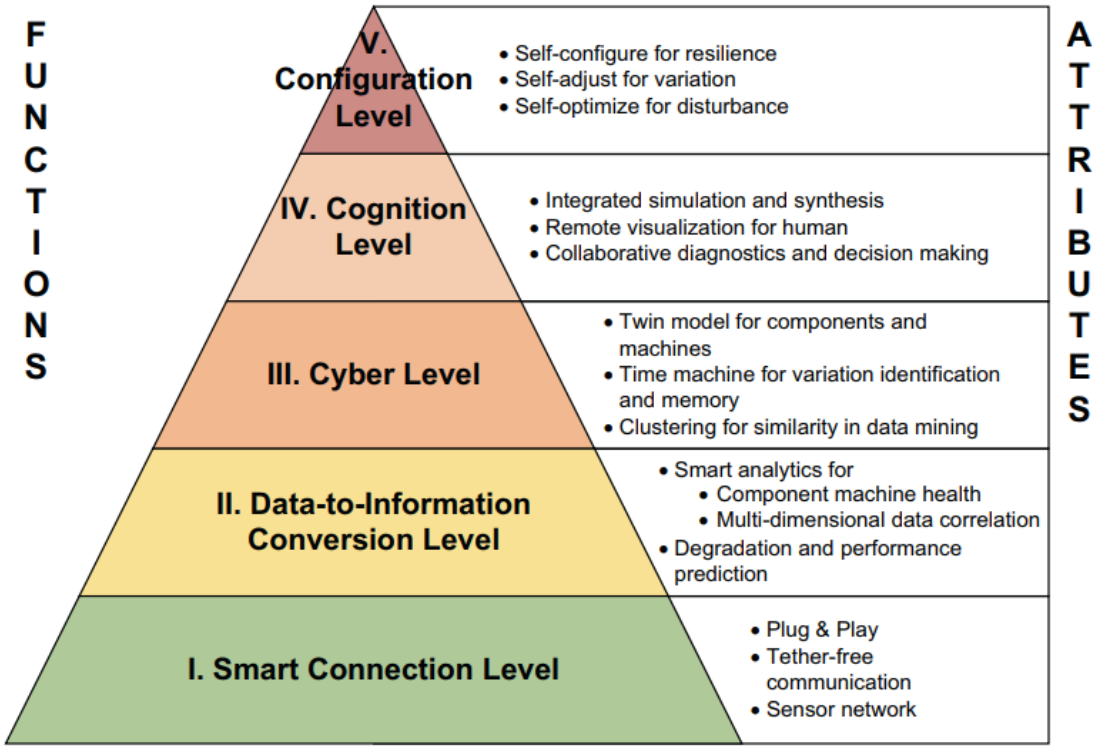

Figure 2.4. *5Cs of Cyber Physical Production Systems [19]*

.

based on so called 5C-architecture. Like shown in Figure 2.4, it consists of connection, conversion, cyber, cognition and configuration. [19]

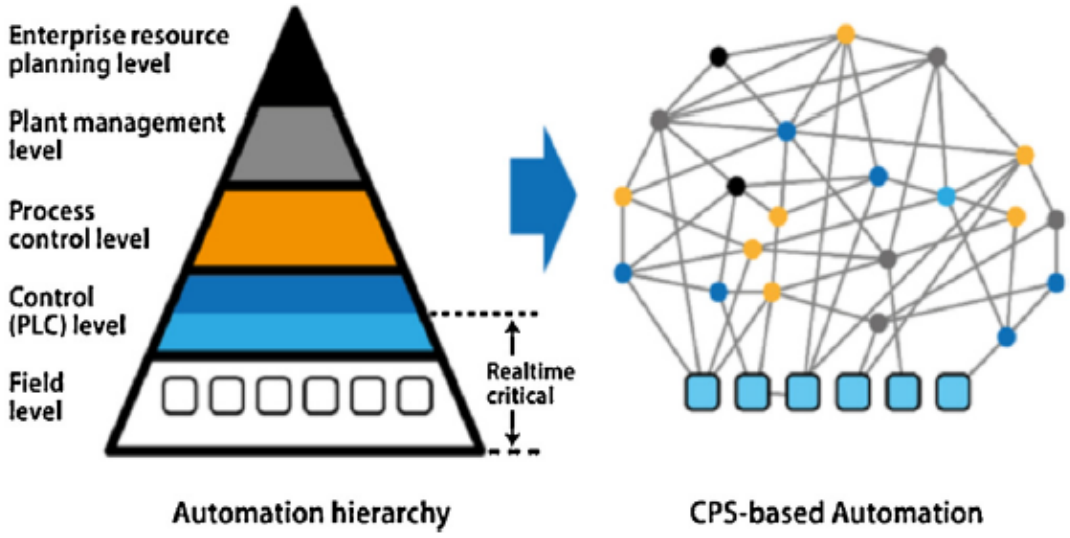

Figure 2.5. *Industry 4.0's changes to the automation pyramid from ISA-95 [4].*

CPPS also partly break the model of the traditional automation pyramid shown in section 2.1 in favor of a more distributed, networked solution. This however does not affect the real-time critical field- and PLC-layers, as high speed control and feedback loops are essential

for production. However the higher levels of the automation pyramid are no longer valid when considering CPPSs. In Figure 2.5 is depicted how the pyramid shaped hierarchy will be broken down to a meshed network of independent actors. [3, 4]

Expectations for CPS are manifold and often very high as they are seen as enablers for new business models and services which would change many aspects of our lives [4]. Some concrete requirements for a CPPS include integrating analytical and simulation-based approaches more than ever, operating sensor networks, handling big data and retrieving, representing and interpreting information while simultaneously upholding strict system security [20].

[4] describes many proceeding technological steps that have emerged on the path to CPPSs, like Intelligent Manufacturing Systems (IMS), Reconfigurable Manufacturing Systems (RMS) and finally Digital Factory. Its goal is to map most of the technical and business processes of a factory to the digital world. An example system of a discrete event simulation (DES) can be seen in Figure 2.6, where several different analytics methods are used to produce information for decision-making.

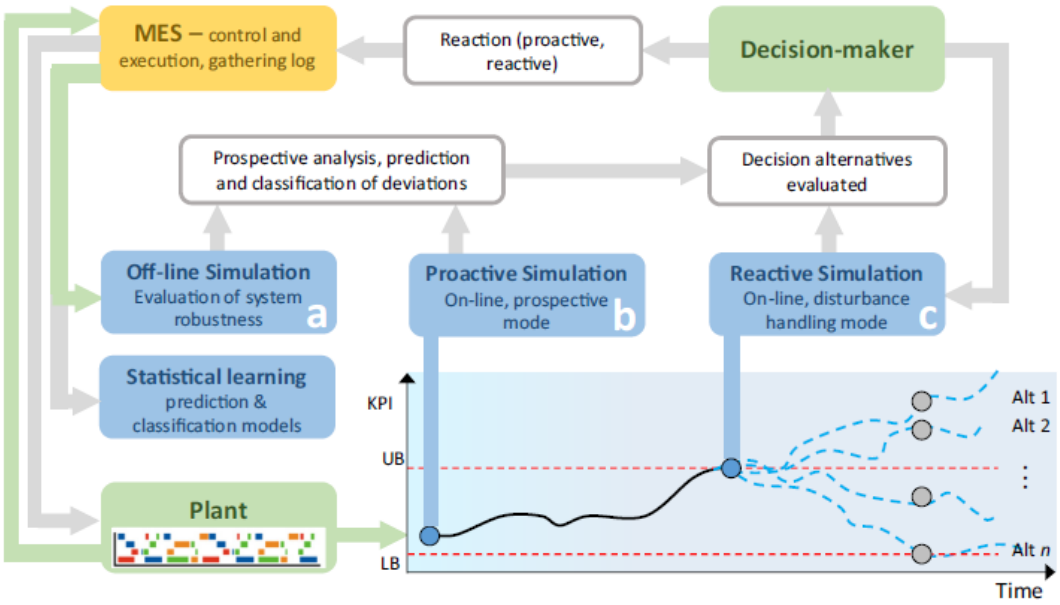

Figure 2.6. *Using analytics in a factory setting to improve decision making [4].*

Finally all the sources used for this chapter mention numerous challenges related to realization of the CPPS vision. The vast majority of these revolve around standardization, integration and communication. [2, 4, 19, 21] In a system based on the ISA-95 (shown in Figure 2.5) pyramid, an entity only needs to communicate with the layers above an under it, in CPS-based automation there has to be a common language or languages, which can be used to communicate globally inside a CPPS.

2.6.3 Smart Factory

Smart Factory, although at this moment just a theoretical concept, is a core element of Industry 4.0. It allows individual manufacturing, resilient manufacturing and augmented operators. The concept is enabled by cyber-physical systems (discussed in subsection 2.6.1) and by Internet of Things. However in an industrial context these both have a counterpart, namely cyber-physical production systems (discussed on subsection 2.6.2) and Industrial Internet of Things (discussed in section 2.4) [2].

In a Smart Factory, IIoT is used to integrate the factory resources with cloud. Cloud is also used for big data analytics, storing data, analyzing and forming decisions. Also humans will be more integrated to the production through possibilities provided by IIoT. Chen et al. describes a smart factory as ”an engineering system that mainly consists of three aspects: interconnection, collaboration and execution.” [2, 22]

3. STATE OF THE ART

In this chapter the state of the art for data analytics and data storage are discussed. The main sources used for this chapter contain industry reports from Frost & Sullivan and the expert interview found in Appendix A.

3.1 Data analytics

As mentioned before, analytics is a catch all phrase for different methods and technologies aiming at producing information out of data. It ranges from simple statistical analysis to modern analytics with complex systems that may involve big data, ML and neural networks etc. Due to the shift towards these new methods, also the whole field of analytics is heading more towards programming instead of using traditional ETL (extract, transform, load) tools with graphical user interfaces (Appendix A). [6, 7]

Frost and Sullivan identified that the most important drivers for implementation of advanced analytics are the following:

- Increased amount of data available
- Increased computational power and storage available due cloud platforms
- Evolving analytical algorithms
- Need for business agility

The increased amount of data allows creating data-driven organizations that can unleash the potential of their data to gain insight to their performance, diagnostics, modeling, forecasting and decision-making. Working with big data sets and models that need to be trained, requires more and more computing power and storage. However not many organizations have the resources to invest into dedicated analytics hardware that may be used now and then. That's where cloud computing steps in: in addition to user friendly environment dedicated to developing analytics, it also provides smooth scaling up and down when required and the user is billed only for resources consumed. This allows companies to start small, innovate, and when they feel confident about their models, train them fast with huge amount of computing power. Also cloud services provide often a very wide range of existing services, which in many cases speeds up development and makes integrations easier. (Appendix A) [8]

Another aspect in advanced analytics has been the emerge of small startups that disrupt the established players in the markets. Often backed by university researchers, these small

companies can with patented algorithms provide customized solutions for specific industries and their applications. [11]

Finally, in the scope of Industry 4.0, organizations need to change their ways of working in order to maximize their gains from the fourth industrial revolution. Some of them include optimizing their production to a whole new level with systems thriving on data. On the other hand intelligent decision-making requires insight provided by advanced analytics. [8]

Frost and Sullivan [8] also identifies three key challenges for advanced analytics: Firstly the lack of executive buy-in and inadequate ownership leads to ineffective implementation and to the lack of cross-functional integration. Secondly a shortage of human talent as the demand is far outweighing the supply. Finally, finding the right mix of tools to fit the organization is hard and finding them requires the right talent, cross-functional integration, customer involvement and asking the right strategic questions.

3.2 Data storage

Until recently, data has been usually stored in massively parallel processing (MPP) databases (Appendix A), which can be described with some simplification as multiple parallel SQL databases with their own compute nodes controlled by a single node responsible for outside connections. Microsoft Azure's example implementation architecture can be seen in Figure 3.1 where the Data Movement Service (DMS) is responsible for moving data across the nodes and running queries in parallel.

Lately the focus has been shifting from these MPP databases to varying data lake solutions,

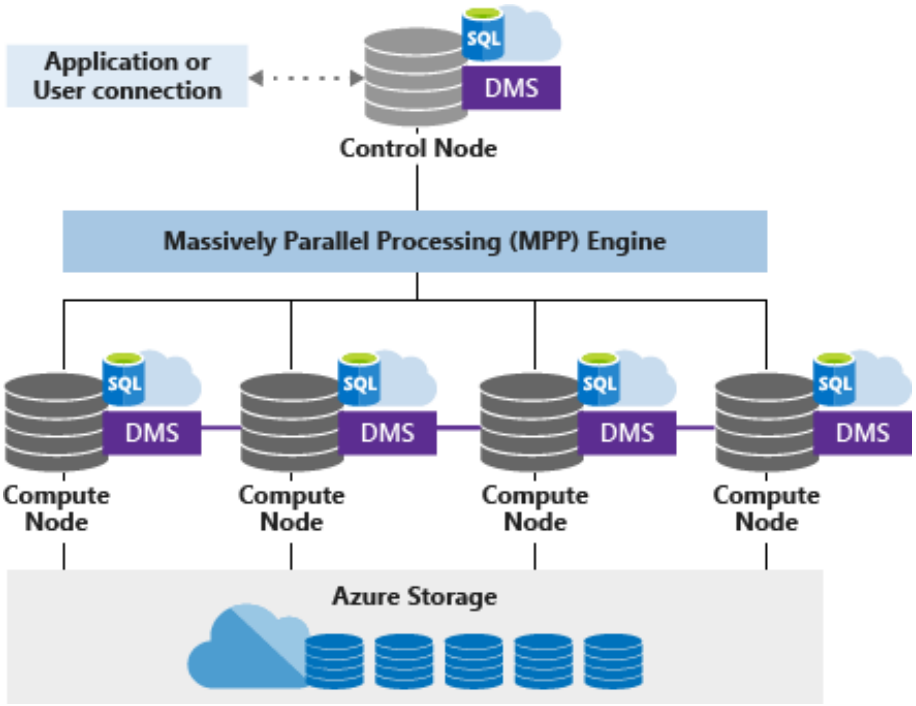

Figure 3.1. *Microsoft Azure's architecture for a MPP Database [23].*

where data is not stored in SQL databases, but instead as blobs. This allows storing bigger amounts of data, as blob storage is much cheaper than SQL-databases. For example storing 100 TB of data in AWS's blob storage S3 would cost around 2400 dollars, whereas storing that same data in an SQL-database would cost around 13 700 dollars (not including traffic costs from accessing the data).

Data lakes and Data Warehouses

In the Figure 3.2 can be seen AWS's comparison of these two technologies. As from the comparison and interview in Appendix A can be deduced, relational databases are still needed despite data lake solutions, as data lakes are better suited for big masses of unstructured data, whereas data warehouses and traditional SQL databases geared towards structured data. One example mentioned in the interview where traditional databases are needed even in systems build around a data lake, was storing analytics results produced from the data lake's data.

Characteristics	Data Warehouse	Data Lake
Data	Relational from transactional systems, operational databases, and line of business applications	Non-relational and relational from IoT devices, web sites, mobile apps, social media, and corporate applications
Schema	Designed prior to the DW implementation (schema-on-write)	Written at the time of analysis (schema-on-read)
Price/Performance	Fastest query results using higher cost storage	Query results getting faster using low-cost storage
Data Quality	Highly curated data that serves as the central version of the truth	Any data that may or may not be curated (ie. raw data)
Users	Business analysts	Data scientists, Data developers, and Business analysts (using curated data)
Analytics	Batch reporting, BI and visualizations	Machine Learning, Predictive analytics, data discovery and profiling

Figure 3.2. *Amazon's comparison between a Data Warehouse and a Data Lake [24]*

3.3 Reference architectures

Reference architectures provide recommended structures for IT products, and they usually embody accepted industry best practices [25]. They seldom can be implemented as is, as they do not take into consideration every organizations unique needs. However, they should be used as a baseline when starting to form a solution. Also the Industry 4.0 report the working group emphasized the need for a reference architecture to provide a "framework for the structuring, development, integration and operation of the technological systems relevant to Industrie 4.0." [16, p. 39]

In this chapter high-level reference architectures are presented both from Solita and the customer's organization for a data & analytics platform. These platforms are designed to be

a central platform for the whole organization, and therefore they should cater various needs that may arise. For comparability, both are represented with Azure components, as that was the composition for the customer's material. However, AWS has matching components for everything.

3.3.1 Solita

Solita's reference architecture shown in Figure 3.3 for data & analytics platform consists of 5 vertical domains and one horizontal domain. The basic flow is left to right, beginning from Business Data Sources, going through Data Ingestion , Data Storage & Enrichment & Publish, Information Interaction and finally ending up in Channels where the data is published. On the single horizontal domain are Development & Operations and Cloud Governance.

Business Data Source's are quite self evident: they are the data sources that need to be connected to the platform. They are very heterogenic, as they may contain IoT devices, excels, various databases, text files etc. Also their number is usually growing, as more and more data is collected.

Data Ingestion domain is the ”lobby” for the platform: all data enters there. The heterogeneousness of data ingested is reflected to this layer as well. For batch data a process for ETL is required, IoT devices need an event hub for the MQTT data and many other softwares are integrated via an API, usually a REST API.

Data Storage & Enrichment & Publish layer is responsible for persisting the data, usually in a data lake or a data warehouse, enriching and publishing it. It also contains all production-ready analytics. On top of the publishing layer provided by it, is built the Information Interaction domain. It provides possibilities for users to get information and interact with the data via visualizations and dashboards. It also allows running experiments with the data in order to create new insights out of it.

The last domain represents all the channels where the results of this process are published to. These can vary from reports in an Intranet to real-time decisions in a manufacturing process.

One special component used in this architecture, is Solita's Agile Data Engine. It is an integrated Data Devops and operating environment, and it helps in building and operating cloud-based data warehouse. It makes the developer's job easier, but is not an irreplaceable part of the architecture. [26]

At first sight, the template contains quite many building blocks and covers many bases, but when starting to implement a new platform, not everything needs to be done at once. For the case in hand, a first iteration could contain only one path from source data to publishing channels. As open cloud services scale very well, this kind of an approach

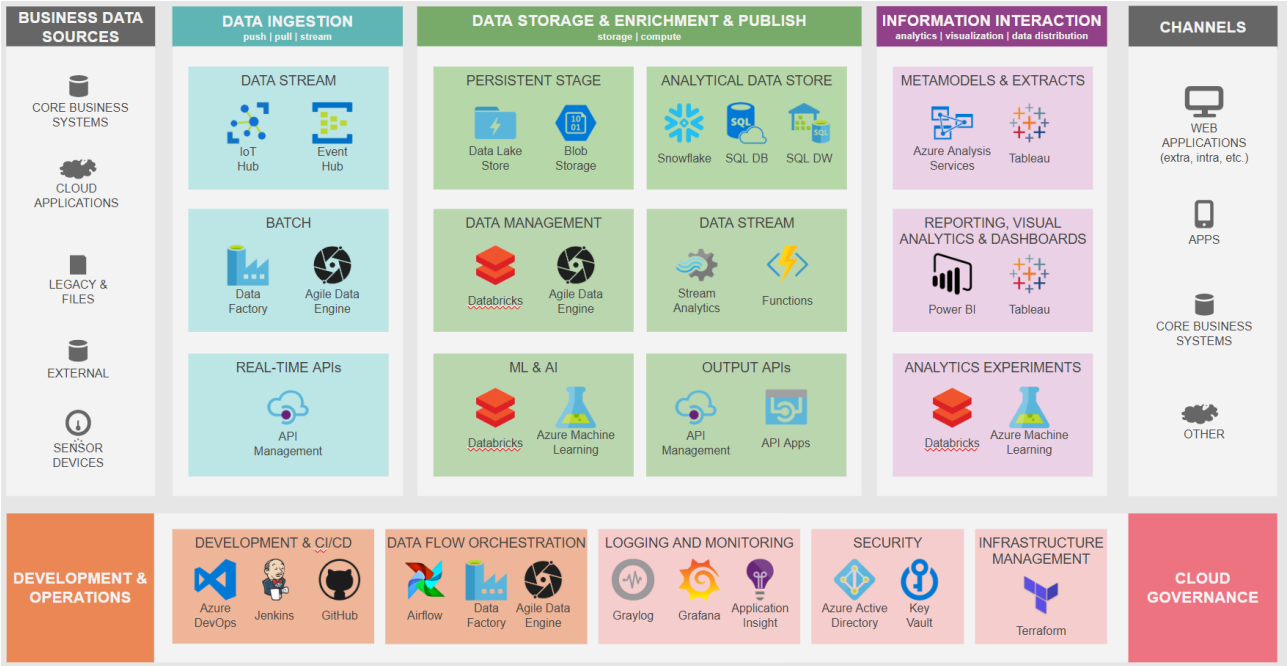

Figure 3.3. *Solita's template for a data & analytics platform*

would also mean that the price for the infrastructure would stay small compared to a full blown implementation.

The original version of this architecture also addresses Data Governance, which in essentially covers the technical, cultural and organizational rules that ensure the platform performs and will continue to perform on the desired level. These may include guidelines on what kind of data should be ingested to the platform, what kind of integrations should be done or how will have access to what part of the system. These are all important topics, but unfortunately not essential part of this thesis' scope and are therefore omitted from this analysis.

Finally on the bottom of the architecture can be seen the tooling that developers and platform operators will use to develop, operate and maintain the platform. As this reference architecture is based on an open cloud platform, the tooling used can vary a lot, as there are little limitations from the platforms side. Also as these platforms are widely used, a lot of open source implementations exists.

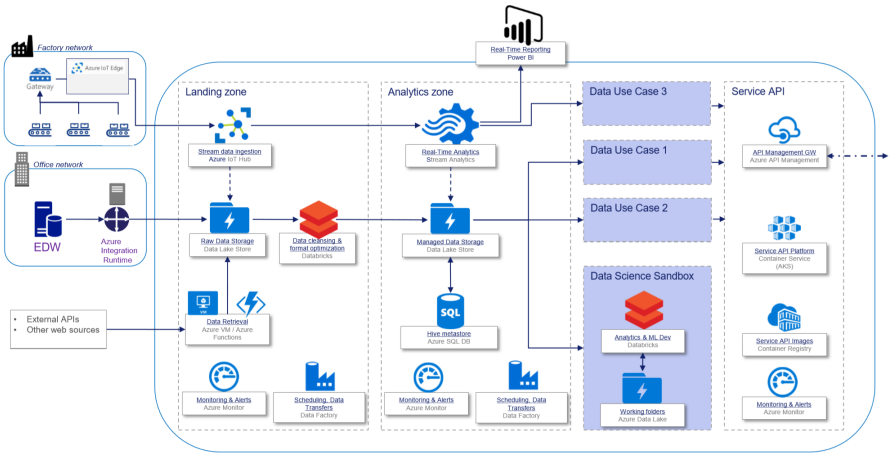

Figure 3.4. *Customer template for their data & analytics platform*

3.3.2 Customer

The customer's reference architecture shown in Figure 3.4 is very similar to Solita's one. Some differences appear specially in the amount of detail: Solita's model is slightly on a more general level, where as the customer's one is already tailored slightly into their organization's needs, just as it should be.

Another interesting difference is inclusion of development and operations tools in Solita's model. This difference is probably caused by the fact that Solita has experience in running these kind of platforms, and therefore has experience on selecting good tooling.

A final difference between these two models is the amount of non-Azure products used: Solita's model include Snowflake (subsection 5.3.3), Tableau and of course the Agile Data Engine that was mentioned before. These differences may be caused some policy decision at the customers side (use only native Azure services).

4. DEFINING REQUIREMENTS

In this chapter the requirements and expectations of the case organization are listed and grouped. Also the scope limitations are defined here.

4.1 Customer requirements

A global problem affecting our society is the high amount of food waste we produce. Up to 30 % of the food produced ends up being thrown away. This is one of the key problems that the customer's Head of Digital Transformation mentioned in Appendix C which company has determined to reduce in the future. However achieving this goal is a long road, and they have decided that adding more transparency to their production operations and starting to build maturity in themes mentioned in Industry 4.0 are the first steps towards it.

The customer's short time goal is to provide more visibility to the manufacturing process and to start understanding it more. Currently too much information is conveyed to SAP by humans or non-automatic means, which leads to information being out-of-sync or no information at all. Also the customer would like to give their employees better feedback on how they perform their jobs, as at the moment is virtually impossible for them to understand the non-immediate consequences of their actions. All this requires that data needs to be collected from the process, it has to be analyzed to refine it to information and finally that information needs to be displayed to the relevant persons.

To prepare for that project, I was contracted with this thesis to explore and to analyze the possibilities available and to lift up important topics they should address before moving forward.

4.2 Scope

Due to a very fast evolving and broad topic, the scope of this thesis was limited with several factors that were revised during the writing process.

4.2.1 Data gathering

In this thesis the data sources taken into consideration are limited to one production line using Siemens PLC. In their current situation, which was investigated with an interview included in Appendix B, very little data is gathered, approximately under 2GB per month for the line in question. In the interview it was mentioned that not all current PLC are Ethernet ready, but to limit the scope of this thesis, an assumption was made that all relevant

devices can be upgraded at least to a level where OPC can be used. The customer was also interested in providing context data for the data gathered from the PLC, but the main target was to identify different product and architecture alternatives and their pros and cons. Also the topic of device management should be covered at least briefly, as managing masses of devices manually or with lacking systems is very labour heavy.

4.2.2 Data pipeline

A key priority identified in the assignment given was managing loss of connectivity. Otherwise no hard constraints were implemented regarding the data transfer method. The factory in case has good communication connections (fiber optic Internet connection and good 4G coverage). Security is naturally important when dealing with business related analytics with big potential monetary implications and it should be considered in all stages, specially in the data transport between the factory and the cloud. However a security audit is not the purpose of this thesis and it is sufficient if a given technology can be made secure with a feasible amount of work.

Another priority is the scalability of the pipeline. In the scope of this thesis only one production line is considered, but some thought should be given on for example how to scale the given system to include the data of an identical production line in the same location.

4.2.3 Cloud

The domain of cloud services is very dynamic and competing cloud providers are racing to publish new features to the markets in order to gain market position. Specially when considering IoT platform this has lead to a situation, where there are hundreds of platform providers ranging from big tech majors like Amazon and Google, to small innovative startups [27]. Making an exhaustive study of these products and services is nearly impossible. The services and technologies covered in this section were chosen with three criteria: suitability for the customers case, market adoption of the product and the investments and resources of the product developer. A strict criteria was also implemented demanding that the solution provided is cloud native. This means the solution has to be developed for primary for cloud in mind, no on-premises solutions ported for cloud usage are included. This does not exclude hybrid solutions offering offline capabilities from this study.

4.2.4 Derived requirements

Some criteria identified for evaluating the suitability to the customer's case include

- Possibility to support optimization in the future with a feedback time of less than a second
- Possibility to handle heterogeneous data like pictures etc.

- Easy integrability to different systems (import and export data from/to different services located in a cloud environment)
- No data losses when connection problems occur
- State-of-the-art security
- Possibility for running advanced analytics
- Platforms adaptation and future outlook

5. RELEVANT TECHNOLOGIES AND PRODUCTS

In this chapter technologies and products relevant to the domains of this thesis are discussed. The products were chosen with the help of the criteria defined before, customer interest (for example Kepserver was mentioned by the customer in Appendix C), the writers personal experience and literature used as references in this thesis. Criteria used to choose the product in question are mentioned in the respective chapters.

The technologies and products are divided in to three sections: Data collection, Data transport and Cloud. However some large entities like Amazon Web Services, are impossible to fit under one section. In such cases, the relevant components are introduced under the relevant section.

5.1 Protocols

This chapter covers the most important protocols for IoT. There are also proprietary protocols that some products use, but this thesis will focus on the open protocols, because in [16] it was stated that the foundation of CPPS's and Industry 4.0 are open, standardized protocols. These protocols in question were chosen as they are seen as the most prominent IoT protocols in several academic papers, and they are supported by majority of the cloud products and platforms covered in this thesis [28, 29].

5.1.1 HTTP

The Hypertext Transfer Protocol (HTTP) is a Transport Control Protocol/Internet Protocol (TCP/IP) based application-level protocol that has been used in the World-Wide Web since 1990. It has had several versions, and the most current one is HTTP/2, which was standardized in may 2015. [30, 31] HTTP is a client-server protocol and its paradigm revolves around documents. It is very simple but extensible, which may the factor in its long life and high adoption rate.

HTTP traffic can be encrypted using certificates. The most modern protocol for doing that is Transport Layer Security (TLS), which is the successor of Secure Sockets Layer (SSL). This secured traffic is called HTTPS (Hyper Text Transfer Protocol Secure). [32] HTTP is a best-effort protocol and does not provide any Quality of Service (QoS) features in addition to to the underlying TCP. This may be problematic in some applications, where it's critical that all data is transmitted successfully once and only once. HTTP has also a much larger bandwidth requirements and a higher latency, as it is protocol designed to be very interoperable and contains a lot of context information compared to protocols designed for IoT usage. [28]

5.1.2 MQTT

MQ Telemetry Transport (MQTT) is a simple, lightweight, publish/subscribe model messaging protocol published in 1999. It was designed for constrained devices and low-bandwidth, high-latency or unreliable networks. Like HTTP, MQTT is based on TCP/IP. Its design principles are to minimize bandwidth and device requirements and simultaneously provide reliability and some degree of delivery assurance. [28, 33]

Like HTTP, also MQTT can be used together with Transport Layer Security (TLS). This however causes an overhead for the otherwise very light protocol. However this is usually a problem with short-living TCP connections, with long-living connections the TLS overhead should not have a big impact. [34]

5.1.3 AMQP

The Advanced Message Queuing Protocol (AMQP) was developed in 2003 by John O'Hara at JPMorgan Chase. It supports both request response and publish subscribe architecture. It is quite similar to MQTT, but offer far more features. One of these is TLS integrated to the standard, where as MQTT relies solely on security provided by the transportation layer. AMQP also provides better user security with stronger passwords. [35]

However these features come with a price. AMQP header size is four times bigger (8 Bytes vs 2 Bytes), which affects performance specially if packet payloads are small [28]. AMQP is also not as widely supported as MQTT, for example AWS IoT does not yet support it [36].

5.1.4 OPC UA

Open Platform Communications (OPC), now days OPC Classic, was created to abstract PLC specific protocols like Modbus and Profibus into a standardized interface, which could be consumed by HMI or SCADA systems [37]. It is based on a Client/Server model, where client generates requests which the OPC server then fulfills [38].

OPC UA (Unified Architecture) is the successor of the widely adopted OPC. Their biggest difference is that OPC-UA is platform agnostic, where as OPC is based on Microsoft DCOM technology and can therefore only be run on Windows. OPC UA also introduced a modern information security to the protocol, implementing SSL, HTTPS and X.509 certificates. [39]

Another big improvement in OPC UA is the introduction of information models. With these models it is possible to provide context data essential for providing meaningful data analyzing. As standard, OPC-UA implements the model from OPC, but it can be extended with industry standard information models and vendor specific models. [40]

Siemens has adopted OPC UA widely and relies on it also in internal communication between its systems. They state that OPC UA's integrated security mechanisms, vendor- and platform-independence provides the best foundation conditions for digitalization. [41]

OPC UA PubSub

OPC UA PubSub is the newest addition to the OPC UA standard released in February 2018. It establishes a standardized way for implementing a publish subscribe pattern to compliment the traditional server client pattern of OPC UA. One of use cases behind this standard was to allow OPC UA Servers to represent services and devices to cloud based applications, like big data analytics, optimizations and predictive maintenance. [42, p. 4]

PubSub can be used with or without a broker. Without a broker the protocol used is OPC UA UDP, which is usually suited better for communications in a limited network like a factory network. However when using a message broker, PubSub allows using established protocols like AMQP or MQTT combined with a JSON payload for an easy integration with the cloud. [42, p. 4]

An example developed by the OPC Foundation can be seen in Figure 5.1. In it a publisher application is configured with JavaScript Object Notation (JSON) files. After that it forms a connection with an existing OPC-UA server, and begins publishing events to brokers based on that data. The examples used in this case are Mosquitto, an open source broker developed by Eclipse and Azure IoT Hub that will be covered in section 5.5.2 [43].

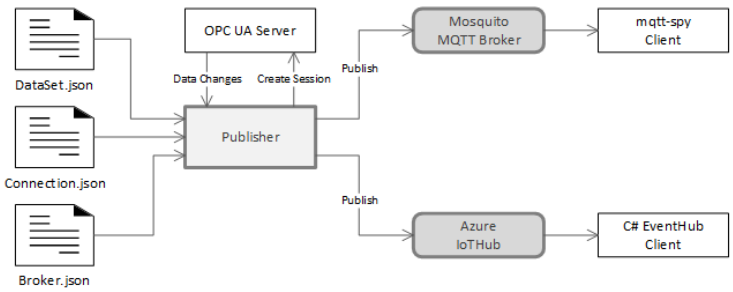

Figure 5.1. *Architecture of a OPC UA PubSub functionality using two message brokers. [44]*

One key feature of PubSub is that it retains the OPC UA data model [45]. This standardized mapping to JSON allows a harmonized data model among all OPC UA data sources and no vendor specific schema is required for understanding the data. This type standardization is one of the key requirements for the realization of Industry 4.0.

5.2 Data collection, Edge and Connectivity

Data collection in the scope of this thesis consists of all activities executed inside the factory to gather, modify, prepare and to store the production data.

5.2.1 Siemens PLC

Siemens Group is a German technology company specialized in electrification, automation and digitalization [46]. Their line of PLCs for industrial automation are called SIMATIC Controllers [47]. They range from basic controllers like S7-1200 to advanced controllers like S7-1500 or distributed controllers like ET 200SP.

Siemens automation systems use their own field bus called PROFINET, which is an IP and Ethernet based protocol for Industrial Internet. Siemens also provides a wide range of add-on modules and software to accompany their PLC products, like Manufacturing Execution System (MES) products and Supervisory Control And Data Acquisition (SCADA) products. [48]

Siemens provides their own implementation for an OPC UA -server that can be used to access server data from Siemens PLCs [49]. However products like KepserverEx are able to read the data directly from the PLC. Siemens's Totally Integrated Automation (TIA) portfolio uses PROFINET for field level communication, but also OPC-UA for vertical communication [41].

5.2.2 KEPServerEX

KEPServerEX is a connectivity platform which aims for being the single source for industrial automation data to other applications. It is the main product of Kepware which is owned by PTC Inc. Kepware has developed automation software since 1995. [50]

KEPServerEX provides a wide range of drivers and plugins which allow support for numerous different protocols. These drivers can be bought alone or bundled into manufacturer specific suites (for example Siemens Suite) or larger suites like the Manufacturing Suite which contains drivers for wide range of automation hardware and open standards like OPC and OPC UA. [50]

Among the numerous possible drivers and plugins is KEPServerEX's IoT Gateway -plugin. It allows using MQTT, REST or ThingWorx's proprietary Always On -protocol to stream data to desired endpoint. It supports common and custom message formats, multiple application connections and a remote configuration for all agents. [51]

In Figure 5.2 is depicted how KEPServerEX alongside with its IoT Gateway integrates data sources like PLCs with cloud based actors like cloud analytics. In addition to basic data relaying, KEPServerEX provides also some basic edge computing features, like aggregating and arithmetic expressions as well as functionality for load balancing and redundancy. [52]

Kepserver does not have any device management software or service, but it provides a Configuration API. This allows configuring Kepserver itself as well as the device configuration. However it is just an interface to do configuration, Kepserver does not provide any software, that would do the actual management like keeping track of assets or creating new configurations automatically. [53]

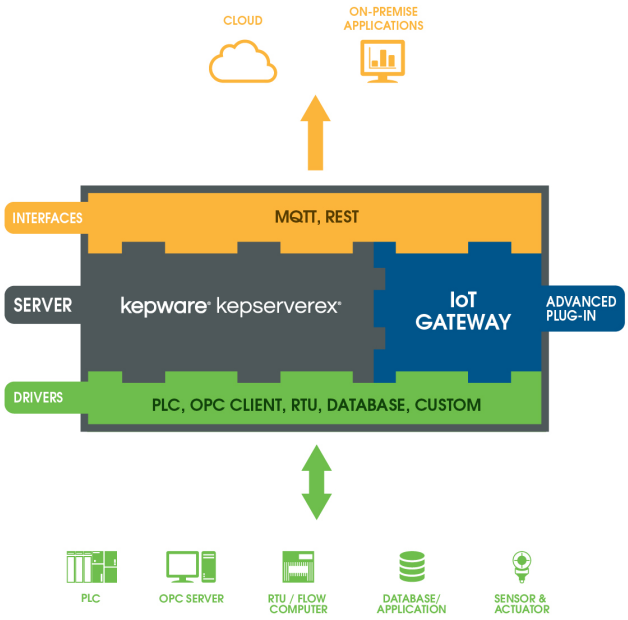

Figure 5.2. *Kepserver's architecture with IoT-Gateway included. [51]*

5.2.3 AWS IoT Greengrass core & IoT Device SDK

AWS IoT Greengrass core is a software developed by Amazon and it is designed to be used in devices connected to their cloud platform Amazon Web Services (AWS). It allows devices to use AWS's resources when required, but also to function if taken offline, or to use resources in local network if it is more suitable in the situation [54]. AWS IoT was chosen as a relevant technology due AWS being the biggest or at least one of the biggest cloud providers, and they have a comprehensive IoT related offering [55].

AWS IoT Greengrass core is a part of the AWS IoT Greengrass ecosystem, however it is not run by Amazon in AWS, instead it is meant to run on a local edge server or a gateway. It functions as a service provider for local devices, that provide security, connectivity to

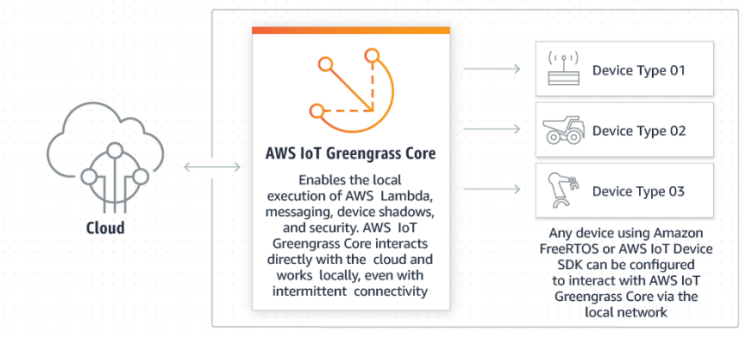

Figure 5.3. *AWS IoT Greengrass core and its role between the cloud and devices [54]*

AWS and other devices, execute lambda-functions locally or even run machine learning models. This role is visualized in Figure 5.3 [54][56]

AWS IoT Greengrass also supports OPC-UA out of the box. This means a Greengrass core device is able to read and monitor values from an OPC-UA Server, process them and relay them to a cloud service. [57] An example implementation with AWS Greengrass and Beckhoff PLC can be found in [58].

AWS has a program for certifying hardware that is tested to be compatible with their services. Currently there are listed 21 different edge servers that are certified to work with Greengrass core from several vendors including Lenovo, Dell and HP. [59]

5.3 Cloud Computing in General

In this thesis cloud is used often as the synonym for cloud computing, which is defined by NIST as following: ”a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction” [60].

We also categorize the available cloud computing services to two categories for simplicity and because they share many similarities explained later: products and open platforms.

Both for products and open platforms, comparing pricing is very hard. The key reason for this is very heterogeneous pricing metrics, where one service may bill based on amount of request, the other may count the amount of data or have even a fixed price. Products may have in some cases a simpler billing model, but these are not as public as for open platforms and may even depend a lot on the customer. Also products often have a rougher pricing tiers, which might lead into situations where one has to pay for more resources than actually uses. (Appendix A)

5.3.1 Multi-Cloud

A traditionally multi-cloud referred to an architecture, where an application was run on several data centers of a single provider. However this is now a standard feature in all major cloud services and is available for most services. The emerging trend in multi-cloud is nowadays using resources from different providers, e.g. AWS and Microsoft Azure, and this is what in this thesis will be referred to with the term ”multi-cloud”. [61]

The main advantage in running services across two or more different cloud providers is increased robustness, financial gains from different pricing models or simply the availability of a certain lucrative service or application that the current provider does not provide. [61]

A multi-cloud model is especially well suited for microservice based solutions, as they consist of numerous loosely couple services. Therefore in most cases they can be quite easily

run in different cloud environments. This also allows using features from separate cloud providers and provides more freedom when selecting what components to use. However it has to be noted, that often transferring data inside a cloud platform is priced cheaper than transferring data in and out of the platform. [62] (Appendix A)

5.3.2 Serverless

One definition for serverless is the following: ”Serverless computing is a form of cloud computing that allows users to run event-driven and granularly billed applications, without having to address the operational logic.” AWS Lambdas and Azure Functions are examples of Function as a Service (FaaS) -applications and serverless. However they rely on APIs that are incompatible with each other and also with other FaaS. Therefore serverless frameworks are emerging to abstract these APIs. [63]

A confusing example is a product called ”serverless”, which allows developing functions and then running them on AWS, Azure, Google Cloud Platform or on Kubernetes. These kind of services allow realizing multi-cloud solutions easier, as there is no need to implement each vendor specific API for each use case, instead the serverless framework will provide a single API that allows running the software on any cloud platform. [63, 64]

5.3.3 Snowflake

Snowflake is a software product developed by Snowflake Computing Inc. It was chosen to be covered in this thesis because it was identified as a modern solution specially for cloud based data warehousing in Appendix A. Its main principle is to separate computing from storage. In AWS ecosystem this means that the storage is handled by S3 buckets which are then operated by EC2 virtual machines. These can be scaled up or down individually providing high granularity flexibility. An example use case could be scaling the computing up for the duration of running a complex monthly summary report. [65]

One of Snowflake’s key advantage is its ability to load and optimize both structured and semi-structured data including JSON and XML in a manner that allows querying it with SQL. Another useful nature for a cloud based data warehouse is Snowflake’s ability to share data easily with different users and even organizations without duplicating it and still enforcing secure access control. [65]

Snowflake is a SaaS-service and priced by usage, naturally computing and storage are billed independently. As a service it eliminates a lot of the traditional data warehouse admin work, including infrastructure management, optimization and data protection. At launch, Snowflake was only available in AWS, but in September 2018 Snowflake announced that it will be generally available also in Azure. [65, 66]

5.4 Cloud Platform Products

Many companies have released in recent years products trying to provide a platform for managing IoT devices, handling the data produced and executing analytics. For this thesis we have tried to select examples from major vendors, as they are more likely to have the resources to produce a comprehensive product and provide future support for it. Also the domain knowledge of the company was valued, mostly by case examples provided on their website. The selected vendors and products are only a subset of the products available.

All platforms mentioned in this chapter offer ”Platform as a Service” (PaaS) -services for running your own applications, but some of them offer also some ”Software as a Service” (SaaS) -services, like analytics or security auditing. Majority of the products chosen are based on an open source software called Cloud Foundry.

5.4.1 Cloud Foundry

Cloud Foundry is an open source (available at https://github.com/cloudfoundry) platform for running applications in the cloud. It is managed by a foundation, which platinum members include companies like Vmware, Cisco, IBM, SAP and their gold members include companies like Google, Microsoft, Ford, Volkswagen, Huawei and Allianz. Therefore it is safe to say it is backed by very major players in the technology industry. [67]

The platform includes an application runtime for running applications, a container runtime for running the containers and BOSH - Cloud Foundry’s toolchain for release engineering, deployments and life-cycle management of large scale distributed services. [68] Cloud Foundry does no recommend to run your own foundry, but to use a certified providers platform. However it is also possible to run your own foundry. Pivotal also offers their Cloud Foundry based product Pivotal Cloud Foundry for self hosting. It can be hosted on several service providers like AWS and Azure. Therefore cloud foundry is not targeted mainly towards companies looking for an platform, but more towards partners using it to build their own platform for their customer companies. Many of the following platforms introduced are based on this software. [69]

One of the less customized and tailored version is provided by Pivotal which can be used as an example how companies build on top of Cloud Foundry. Pivotal’ cloud foundry revolves around 4 main services: Application Service, Container Service, Service Marketplace and the upcoming Function Service, shown in Figure 5.4. Application service provides a runtime applications, Container Service provides a container runtime for containers like Docker and the upcoming Function Services will offer similar services like AWS Lambda and Azure Functions [71]. These services sound very similar to the ones defined by the basic Cloud Foundry.

However what differentiates Pivotal’s version from the open source projects, is the Services Marketplace [72]. It provides developers with services developed by Pivotal or other

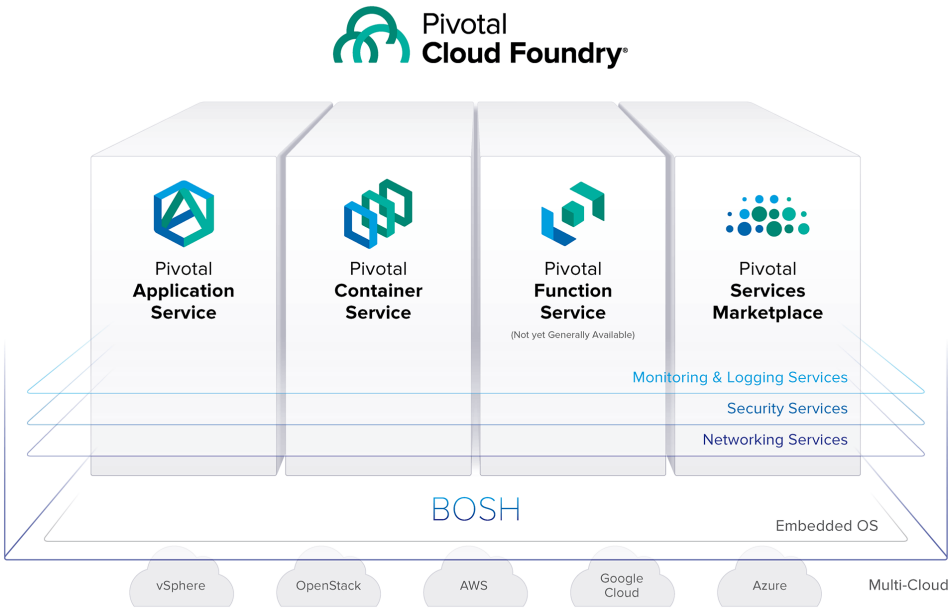

Figure 5.4. *Pivotal Cloud Foundry's architecture [70]*

companies. It even includes connectors to some Azure, AWS and Google Cloud services, although only to a very limited subset. The open source version does not have any of these, although some boilerplates may be found in GitHub as open source projects.

5.4.2 Mindsphere

Mindsphere is an "cloud -based IoT operating system" developed by Siemens. It runs fundamentally on AWS or Azure, but is a closed source product built on top of Cloud Foundry. It provides some applications, but mainly it is an open PaaS environment. It offers secure connectivity to variety of things, including plants, machines, enterprise applications and legacy databases, but the most important functionality is the plug-and-play connectivity for Siemens products. [73] Siemens has invested massively in software development and has invested 10 billion dollars into acquiring software companies since 2007 [74].

Mindsphere provides some basic services, like asset-, user- and fleet management, some visualization and data exploration tools, developer tools, connectors and one service for edge analytics and one for predictive learning. In addition to this, there are a applications provided by Siemens and some other third party companies, that can be used in Mindsphere. Some of these applications are free but most of them are paid. In October 2018 there were 23 applications available, however in January there were only 14 left. At least applications developed by Accenture seem to have disappeared during late Autumn 2018, which could be an indication of diminishing interest towards the platform by application developers. [75]

Mindsphere's device management features are quite on par with the competition. They

MindAccess IoT Value Plan	Small	Medium	Large
Pricing	€3600	Contact Us	Contact Us
Productive Tenant	✓	✓	✓
Access to MindSphere Store	✓	✓	✓
Fleet Manager[1]	✓	✓	✓
Users	50	150	500
Subtenants	10	40	100
Asset types	5	10	50
Asset instances	50	250	1000
Connected Agents[2]	10	25	100
Data ingest rate[3] (time series)	2 KB/s	10 KB/s	100 KB/s
Cold storage in total	60 GB	300 GB	3 TB
Data ingest via MindConnect IoT Extension, per month	5 GB	5 GB	5 GB
File storage	50 GB	100 GB	500 GB

Figure 5.5. *Siemens Mindsphere's IoT Value Plan pricing [77]*

provide features for mass provisioning new devices, monitoring them, setting alerts and over-the-air software updates. However their mass provisioning only works with uploaded .CSV files and has no API. [76]

Pricing of the Mindsphere platform and services is not public except for the smallest subscriptions as seen in Figure 5.5. Mindsphere does not offer any free tiers or demo accounts for testing or academic purposes.

The prices for services and their upgrades in Mindshere range from free to tens of thousands per year. For example "Predictive learning" is starting at 16960€/year. For 3rd party applications the provider will decide and handle billing matters themselves [78].

One of Mindsphere ecosystem's key value is the ability to connect Siemens PLCs and automation systems to it effortlessly. For this Siemens provides physical connectivity modules and two software extensions. The first software extensions can be used directly in modern Siemens PLC, which means no additional servers or devices are required. The second software extensions is a normal software developed in the C language and can be run in Linux, Windows or Mac environments. Finally the two physical devices can be connected to the automation system using OPC-UA or Siemens' own field protocol. They

can only send data to Mindsphere and their datasheets do not specify further how this is done. [79]

Mindsphere's offering may be tempting to customers using already a full Siemens ecosystem. However that may increase further the dependency to a single vendor as these kind of platforms lack interoperability with other platforms [80].

5.4.3 Predix

Predix is a platform for digital industrial applications. It is a product developed by General Electric (GE) which is an industrial titan with revenue of 122 billion dollars in the year 2017. [81] Their main selling points for Predix are that it is build around an idea of an "asset-centric digital twin" and that it provides a comprehensive edge-to-cloud architecture and SaaS applications. They claim that they have invested over a billion dollars in the development of Predix. [82]

Predix is based on cloud foundry like all the other services, but has build a lot of services on top of it. To be accurate, Predix is not only build on top of the open source cloud foundry, but on top of Pivotal's version of it. [83] Predix however offers more customized services engineered towards industrial usage that Pivotal, like Edge computing, Data Management, Analytics, Security etc.

The pricing model of Predix consists of tiered and fixed prices, where the general use of Predix is based on the used memory, whereas the use of Services, Analytics and Apps have ether a tiered or a fixed pricing. For analytics models price usually range from tens of dollars to couple hundred dollars a month, whereas services are usually priced by tiers relative to usage. [84] Predix' also provides custom pricing for bigger customers and large scale solutions ranging from hundreds of thousands dollars a year to millions of dollars a year [85]. For development Predix has a free tier, which is limited to 4GB of memory, 10 instances and limited access to certain services. In addition it provides a 60 day Predix Free Trial Offer that gives full access to Predix for a shorter time. [86]

Predix marketplace is very comprehensive and at the time of writing contained 55 different services (6 marked as "soon" and therefore not available yet) and over 100 different analytics models [84].

Predix Edge

Predix Edge is GE's solution for IIoT applications that require edge computing. It was ranked in 2017 by Frost & Sullivan as a leading solution for edge analytics [87]. Predix Edge allows running software and analytics locally and provides an Edge Manager that provides various device management functions like provisioning new devices, monitoring them and pushing updates to them. [88]

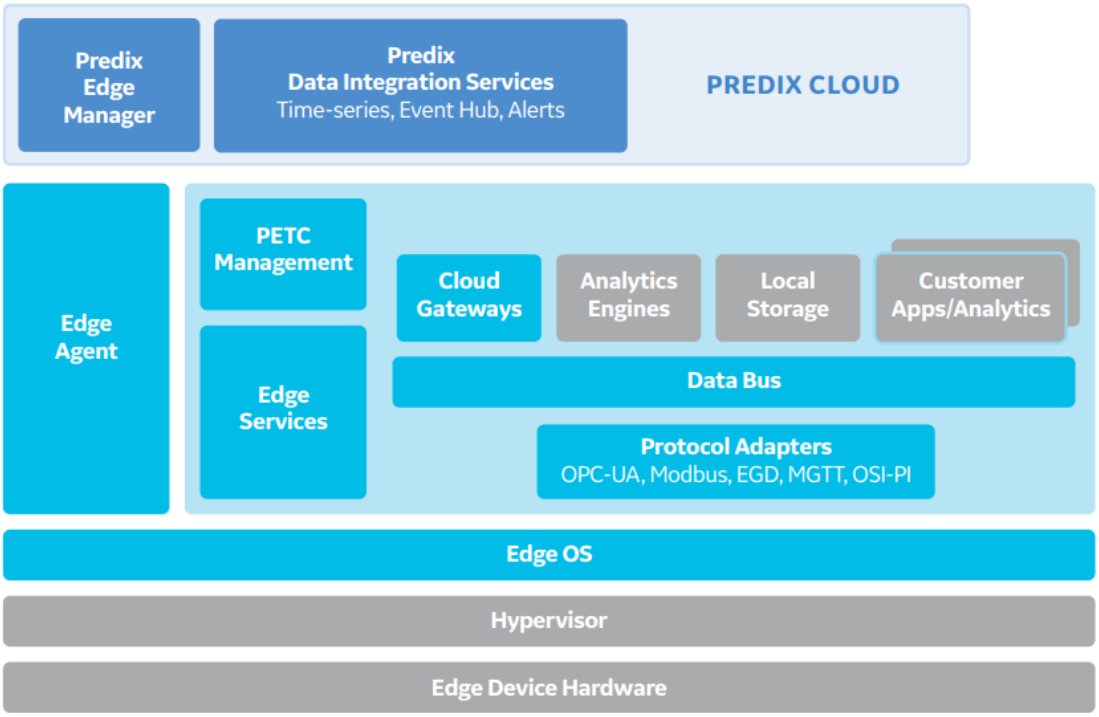

Figure 5.6. *Predix Edge's architecture [88].*

The Predix Edge is based on a hardened Yocto Linux OS that manages the edge devices. It ships with adapters for OPC UA, Modbus, OSI PI, MQTT and EGD. It can also run Docker containers developed in C, C++, Java, Python or Node.js. It can be run on a VMware virtual machines and GE will start providing hardware with preinstalled Predix Edge in 2019. [88]

The Predix Edge's software stack is shown in Figure 5.6. GE also provides a 30-day free trial for their Edge Manager [89].

5.4.4 PTC ThingWorx

PTC Thingworx is an IIoT Platform owned by PTC, which also owns Kepware. In Finland they are represented by Elisa, who offer Thingworx as their IoT-platform[90]. Thingworx can be run on both AWS and Azure as well as a hybrid-solution [91]. However Thingworx has partnered with Azure and aiming to provide more integration to Azure services [92].

Thingworx has a very large sample collection and focuses heavily on graphical interfaces instead of programming. It was also ranked highest in the latest Gartner's magic quadrant for IIoT platforms which was partly due the PTC Marketplace [93].

Thingworx is made of different modules. The heart of the platform is called the Foundation. It provides connectivity and allows users to create and deploy IoT solutions. The Foundation is model-driven and based on ThingModel, Thingworx' model for assets (or things) and it

is consistent through the whole platform. A key feature for the foundation is the ability to design applications rapidly without a need for manual coding and minimal integration work. [94]

In addition to the Foundation, Thingworx has couple of key modules: Analytics and Industrial Connectivity. Analytics provides real-time pattern and anomaly detection, predictive analysis and contextualized recommendations. It is also advertised for user friendliness and easy-to-use tools that eliminate the need for developer or user expertise in data modeling, complex mathematics or machine learning. The second core module, Industrial Connectivity, consists of products and solutions provided by Kepware, better explained in subsection 5.2.2. [94]

Thingworx offers a free trial to their platform that for the most parts last for 120 days. Some elements require periodical reactivation during this period, for example ”Manufacturing Apps” every 30 day and Industrial Connectivity -module requires a restart every 2 hours. Their augmented reality studio ”Vuforia” has also an 30 day trial period. [95]

Thingworx has also device management features including OTA software updates, device monitoring, alerts and mass provisioning [96]. However in the latest version these features have been mostly deprecated and moved to their new service ”Asset Advisor” [97]. Only the possibility to send OTA software updates has not been migrated to the new service.

PTC Marketplace

PTC Marketplace contains a wide range of different tools and products that can be used in Thingworx. They range from an Azure IoT Connector, which allows devices running Azure IoT SDK to send data directly, or via an Azure IoT Hub, to Thingworx, to a complete solution for a farm’s data. From all available tools and products, roughly 1/3 are supported by PTC, 1/3 are supported by a partner of PTC and the rest are unsupported. PTC Marketplace is not transparent about pricing. There are no prices available online. [98]

Thingworx’ Edge Computing

There are very little information about Edge capabilities provided Thingworx. They provide however an SDK and a WebSocket-based Edge MicroServer (WS EMS) to provide TLS encrypted connectivity to Thingworx with their AlwaysOn-protocol. Nether of these however provide edge computing possibilities. The SDK is just a library to handle secure communication, and the WS EMS a gateway that can also handle device configuration with Lua scripts. [99]

However, Frost % Sullivan ranked Thingworx as a strong vendor for edge analytics [87], trailing GE’s Predix with just a small margin. This is probably due projects done with separate edge computing vendors or custom solutions. During the writing of this thesis I left contact requests to Thingworx to clarify their edge computing offering, but got no response back.

5.5 Open Cloud Platforms

Open cloud platforms like Azure and AWS offer numerous Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and even some Software as a Service (SaaS) services with scalability and flexibility. They also provide numerous tools and SDK's to help development, but do not offer domain specific insight like some IIoT platform providers may provide.

For this thesis, Microsoft's Azure and Amazon's AWS were chosen as AWS is the market leader by a large margin, and Azure is the fastest growing competitor[100]. Other relevant providers are Google Cloud Service and IBM. Google opened in summer 2018 a data center in Hamina, Finland [101]. This allows storing some data only inside the borders of Finland and of course a lower latency to assets. IBM on the other hand is a strong contender in machine learning, specially their Watson project has gathered merits i.e. in the medical field of oncology [102, 103].

Although pricing for open cloud platforms is public unlike for the IIoT platform products, comparing the pricing may be really hard due heterogeneous metrics talked in Appendix A. Also when planning a new software, it can be hard to estimate for example how many database queries it will produce under a normal load.

5.5.1 Amazon Web Services

Amazon Web Services (AWS) is a secure cloud platform, which offers computing power, storage, content delivery and many other functionalities. Its first service, Amazon S3 was launched over 10 years ago in March 2006 but in the last 12 years it has grown massively and it has now days over 100 services and customers all over the world. [104, 105, 106]

AWS includes a very generous and comprehensive free tier, where some services have a permanent free usage for a limited amount of usage, whereas some services have a free tier only for 12 months. Few services also have trial periods with varying limits. [107]

AWS EC2 and S3

AWS EC2 and S3 are part of the basic building blocks of Amazon's cloud. EC2 is an abbreviation for Elastic Compute Cloud and it provides virtual machines that can be scaled easily by adding more machines parallel or increasing the performance of current instances.

Amazon Simple Storage Service (Amazon S3) in an object storage (or blob storage). It is a simple way to store bulk data and it can be used as a data-lake. The S3 is very well integrated into other services in AWS, including other storage services like Amazon DynamoDB and Amazon Glacier and computing services like AWS Lambda.

AWS Lambda

When using EC2, one instance being handled is basically one virtual machine, with AWS Lambda you can run directly code snippets (or functions) without worrying about the infrastructure behind it. Lambdas are triggered by events that they usually respond to, which can range from HTTP requests to events in other AWS services like databases or storage. [108]

AWS IoT Core

IoT Core is AWS's cloud IoT platform. It provides secure communication, data processing, routing and device management for IoT-devices. It is a part of a bigger AWS ecosystem for IoT, containing services and software like FreeRTOS and AWS Greengrass described before, and services like AWS IoT Device Management, AWS IoT Device Defender and AWS IoT Analytics described in this chapter. [109]

The main features for the IoT Core are the following:

- Device Gateway
  - Manages connections to devices, supports currently MQTT, Websockets and HTTP 1.1. Is fully managed and and scales automatically, supports over a billion devices.
- Message broker
  - Brokers messages from and to devices, supports multiple messaging patterns and allows a fine grained access control. Is fully managed and scales automatically.
- Authentication and Authorization
  - Ensures all traffic is authenticated and encrypted, supports multiple authentication methods, including X.509 certificates and Amazon Cognito, Amazons own identity management. Can also provide temporary AWS credentials to devices so they can use other AWS services like DynamoDB or S3.
- Registry
  - Establishes identities to devices and tracks their metadata, for example what units does the device use when reporting data.
- Device shadow
  - Persists the state of a devices and allows devices to predefine their future state.
- Rules engine
  - Evaluates inbound messages according to rules and transforms and delivers them to other device(s) or a cloud service(s). Supports also routing to varying AWS services including storage, machine learning, queue handling, alerting and much more.

[110]

AWS IoT Device Management

AWS IoT Device Management is AWS's solution for securely onboarding, organizing, monitoring, and remotely managing IoT devices at scale. It can be used simply for checking the health of a specific device or device group, but it also allows much more, like pushing over-the-air updates to devices or organizing devices in hierarchies and assigning policies to them. It also allows onboarding new devices in bulk, without configuring each device by hand. [111]

Aws IoT Device Management also integrates tightly with AWS IoT Device Defender, which continuously audits and monitors the IoT device network for anomalies. It can be configured to push alerts and trigger actions in various AWS services, including IoT Device Management, if it detects suspicious activity like abnormal traffic amounts or destinations. [112]

Analytics in AWS

AWS provides many tools for analytics. They range from simple building blocks, like S3 and EC2, that can be used in building an analytics solution to SaaS-services provided by third parties like Snowflake or Redis. AWS also provides API-driven machine learning services including video recognition, text analysis, speech analysis and speech production. If you prefer training your own models or even implementing your own algorithm, AWS provides Deep Learning AMIs (Amazon Machine Image) to run your models on EC2 instances and Amazon SageMaker to make ML development and deployment easier.

Amazon SageMaker

SageMaker is a fully-managed platform for building, training and deploying machine learning models at scale. Its purpose is to make implementing ML easier, and it is preconfigured to run containers of TensorFLow, Apache MXNet and Chainer. SageMaker allows developers to train their models with a single click - provisioning and scaling are done automatically. After the model is ready for production, it will be deployed to SageMaker ML -instances which range from small 2 virtual CPU machines to 64 CPU core machines with 16 GPUs and 732 GBs of RAM [113]. All this aims for easier development and deployment of ML models compared to the traditional model where you have to spin up machines with Deep Learning AMIs by yourself to train and tune your models [114, 115].

5.5.2 Microsoft Azure

Microsoft Azure is a set of cloud services, developed by Microsoft. It offers very similar services as AWS and is their main competitor. It launched in 2008 and has 54 separate computing regions worldwide. [116, 117]

Azure's core is built on traditional services like Virtual Machines and Azure SQL Database and Blob Storage, which provide basic functionality for running code and saving data. Like AWS Lambda, Azure has their own service for running code serverless - Azure Functions. Azure also offers a very wide range of other services, including impressive analytics and machine learning services. According to Appendix A, Azure was for a long time front runner in this field, but the competition has since caught on pretty well.

Azure IoT Hub

Azure IoT Hub is a central part of Azure's Internet of Things ecosystem and it can be described as a cloud gateway. Its responsibilities include routing and securing messages, configuring and controlling devices and integrating them with other cloud services. It is fully managed and scales up to millions of devices with an SLA of 99.9 %. [118]

Microsoft provides SDK libraries for devices interacting with IoT Hub. They are available for multiple Linux distributions, Windows and real-time operating systems. Supported languages are C, C#, Java, Python and Node.js. IoT Hub and these SDKs support the following protocols:

- HTTPS
- AMQP
- AMQP over WebSockets
- MQTT
- MQTT over WebSockets

[118]

In addition to the SDKs, IoT Hub also supports direct MQTT connections and can act as a MQTT broker. However it has some limitations and does not implement all functionality described in the MQTT v.3.1.1 standard. [119]

Unlike in AWS, Azure IoT's device management is not an independent service. Instead it is a part of the IoT Hub. It offers also features like tracking device status, configuring them and pushing over-the-air updates to them.

Azure IoT Edge

Azure IoT Edge is an Azure service, that allows running Azure services, 3rd party services or custom code on local machines. These machines can be monitored and managed directly from a cloud-based interface. It consists of three components: the IoT Edge Runtime, IoT Edge modules and the cloud-based interface.

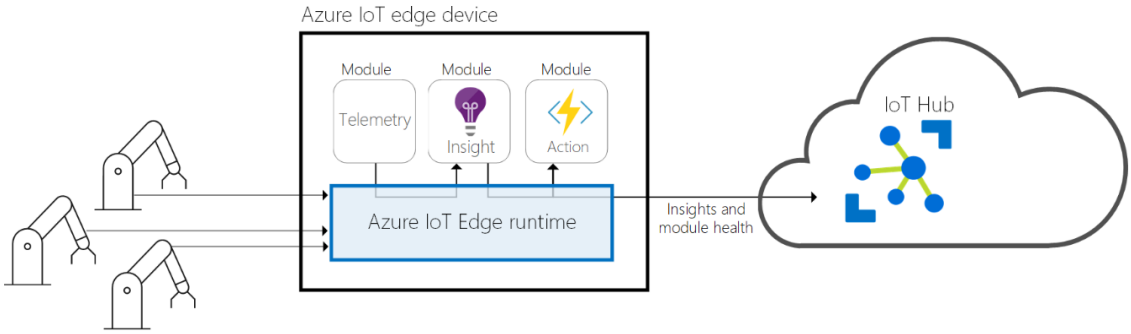

Figure 5.7. *Azure IoT Edge's technology stack [120].*

Like in Figure 5.7 is shown, the runtime is responsible for managing communication and management in the device, including managing the IoT Edge modules. It consists of two components: the IoT Edge agent and the IoT Edge hub. The agent is responsible for managing the lifecycle of modules. However the more interesting part is the Edge Hub. [121]

The Edge hub works as a local proxy for Azure IoT Hub and exposes the same endpoints. It does not handle all requests itself, like authentication requests are simply sent directly to the IoT hub, however it can optimize the amount of connections made to cloud. This way the devices actually producing the data do not need to worry about optimizations. In cases of connectivity loss, the Edge hub will persist messages and digital twin updates locally, and synchronizes them when the connectivity is regained. [121]

Analytics in Azure

Azure provides a state of the art platform for Analytics and Data Warehousing. Some of the analytics and ML services provided are listed in table X. For data storage Azure provides similar alternatives like AWS. Azure Data lake Storage is designed to work well with Hadoop and Spark, where Azure Cosmos DB provides a distributed NoSQL database and Azure SQL Data Warehouse provides a MPP architecture SQL data storage. Also Snowlake mentioned in subsection 5.3.3 is available in Azure. [23, 125, 126]

Azure Stack

Azure Stack is a hybrid cloud platform that allows Azure services to be run in an on-premise datacenter. It allows building applications that have to also run offline or provide a very low

Table 5.1. *A subset of analytic services available in Microsoft Azure.*

Name	Description
Azure Databricks	PaaS for running Apache Spark analytics service.
HDInsight	An SaaS solution for running popular open source analytics frameworks like Hadoop, Spark or Kafka. Can also run R with Microsoft's ML Service.s
Azure Machine Learning	A platform for creating, training, deploying and managing machine learning models. Aimed at providing data scientists a productive environment that speeds development by easy scaling from local testing to cloud-based GPU clusters. [122]
Azure Stream Analytics	Event-processing engine for analyzing high volume data streams like data from IoT Devices. [123]
Data Lake Analytics	On-demand analytics job service for analyzing big data. Works with all Azure data.
Azure Analysis Services	Enterprise grade analytics service (PaaS) that combines on-premises and cloud data sources with enterprise grade semantic data models. [124]

latency. It can be bought through Microsoft's hardware partners that include companies like HP, Dell, Cisco and Lenovo. These systems range from 4 to 12 nodes and are billed according to usage (for systems with connection to Azure) or with a vCPU yearly pricing (for offline systems). [127, 128]

According to [128] it is aimed at scenarios, where the data cannot leave the organization, latency to cloud are too high and connectivity interruptions are to be expected all while there is a need for a consistent development model with Azure. In this light Azure Stack is aimed at quite niche situations and the customers situation does not quite match the niche.

6. EVALUATION

This chapter first starts by summarizing the meaning of the presented theories to the topic of this thesis. After that each domain (data gathering, connectivity and cloud) are discussed and the most relevant discoveries, problems and possibilities are represented. Finally the requirements derived in section 4.2 are iterated through and analyzed which solutions presented in this thesis fulfill them.

6.1 Theory

The drivers behind this thesis are very similar to the challenges IIoT aims to solve, which among others are a better visibility to the company's operations and assets with the help of sensors, software, cloud compute and storage systems. [5, p. 3] Therefore it is important to understand the concepts IIoT connects to, like Cyber Physical Systems, Smart Factories and Industry 4.0.

One of Industry 4.0's most central themes is that it is much more than technical advancement, realizing smart factories or creating cross organization networks. Instead it is as much, or even more, about the transformation of societies together with the industrial field. To gather the benefits envisioned from this change, a lot needs to be done in the fields of research, legislation, education, culture etc. to prepare and to support people in new kinds of careers. These careers will be more specialized than before and include lifelong learning and sometimes even complete re-education during them.

Industrial Internet of Things is just a part of of the Smart Factory concept. That is why it is important to implement IIoT solutions with regard to other parts of the puzzle. Implementing a solution that gathers and visualizes data may be helpful and create competitive advantages, but will also create troubles in the future if it cannot integrate or generate necessary feedback in the future.

6.2 Data gathering

One key requirement for the realization of Industry 4.0 identified in multiple sources is the standardization of communication protocols across value chains spanning through multiple companies. Realistically this kind of protocols need to be open and supported by all major software and hardware vendors. In communication with automation systems the prime candidate has been OPC UA for a long time.

The adoption of OPC UA is very widespread among new automation hardware manufacturers, and it is pretty clear it will be the de facto protocol for accessing data in automation

logic in the near future. However the standard is expanding constantly, so the scope where it will be adopted may vary. A good example of its expansion is the new PubSub expansion.

6.2.1 OPC UA PubSub

OPC UA PubSub's relevance to this thesis is very high, as majority of the solutions explored include processes similar to the content of PubSub. That process is transforming the data acquired from the automation system to a web-friendly protocol like MQTT or AMQP, that uses a publish subscribe model instead of the server client model used by OPC UA. OPC UA PubSub standardizes this process, which allows out-of-the-box implementations for connecting an OPC UA network to a cloud endpoint. It also includes the possibility for using multiple message brokers, which allows for an easy integration with multiple data destinations.

The standard is very new and does not have much support yet, but as OPC UA is endorsed by majority of the automation vendors, it is quite probable that PubSub will be also. Many of them have already released press releases and whitepapers addressing OPC UA PubSub. [129, 130]

6.2.2 Centralized or Distributed system

One key characteristic of a data collection system is it architecture. Centralized systems are often simpler and easier to manage as one entity is responsible for all tasks. This also removes any synchronization problems between nodes. However in high availability systems a single point of failure is not a desired feature. In the planning phase it is important to make informed decisions about the required level of availability and be realistic about requirements: costs start to increase rapidly when required availability starts to approach 100%.

With distributed systems the task are performed by two or more nodes, that can withstand failures of one or more nodes. Traditionally in automation, resiliency against hardware breakages is achieved by duplicating critical components. This is often the simplest choice as it requires just an identical component and some controller to detect the outage and respond to it.

A more recent answer for the demand of resiliency is clustering. A cluster also contains 2 or more, usually at least 3 nodes that can substitute each other when a failure occurs. However compared to traditional duplication, in clustering each node is in use also when no failures are ongoing. This allows for better throughput compared to a traditional duplication where the backup computing is not providing any value when no failures occur. However it requires a load balancer to decide which node to use.

All the products covered for data gathering support redundancy by clustering, except Mindsphere which does not provide information on the matter in the official documents or

on their forums. Kepserver provides an out-of-the-box solution for clustering and has a load balancer, AWS's and Azure's edge solutions can be developed to support clustering and duplication.

6.2.3 Flexibility

The best solution for data collection is also heavily dependent on the source devices. When they are very homogeneous, developing a system is quite straightforward, but as devices, protocols and data formats start to wary, a flexible solution that can be easily modified is required.

One very flexible option is using edge features offered by AWS Greengrass or Azure IoT Edge. They allow development using containers or functions used on the cloud platform. They also integrate the edge and data collection quite tightly to the cloud, which may bring synergy. However the downside is that these solutions are more or less tied down to the platform: the data itself can be quite easily sent to a different endpoint, but this would lead to losing all the advantages of a tightly integrated edge layer.

Finally one relevant topic: update frequencies. Modern machines provide enormous amounts of data as a variable can be read hundreds times per second. Often there is little benefit in sending all of this to the cloud analytics. However it is often pretty hard to decide beforehand what needs to be read on what intervals so an adaptive system should be implemented. Usually this kind of non-static data aggregation falls under the topic of Edge Computing discussed a bit later.

6.2.4 Using MES to Gather Data

Gathering data from automation is very interwoven with a possibly existing MES-system, as usually they already gather all, or at least the majority of the relevant data. The arising question is whether we should collect all of the required data with a MES component and redirect that to the desired analytics platform, or implement an independent service responsible for that. Providing a definitive answer to this question is impossible in the scope of this thesis but would be even if with vastly greater resources very hard, as it would require vetting a large amount of MESs available. Some questions include the relaying delay, robustness and flexibility, as MESs are products and often not that easily modified if the data collection requirements for analytics change. Specially when sensors that are not connected to the automation system are added, the integration to the MES data collection can get problematic. An example of a system that could be complex to integrate into a MES system is a picture, video or audio recognition system, that uses a local model to make instant decisions but uses cloud compute for continuous improvement of its local model.

One possible solution for eliminating a dependency to a MES could be using a single source of data for both the MES and analytics. In this pattern a specialized product (like

KepserverEx), or preferably something using the OPC-UA PubSub architecture, would be used as a data gatherer, which would then integrate both to the MES and Cloud directly. This would allow majority of the data to be sent through MES with enrichment, but provide an easy and out-of-the-box solution for data not supported by the MES.

6.3 Edge Computing

In the requirements it was identified that there will be needs where the required analytics feedback response time is under a second. For these kind of scenarios a round trip to cloud and the processing delay there will become a challenge, specially if there is a strain on the network or some problems in the cloud service (like using a fallback region). Therefore it can be deemed that some kind of edge capabilities are required.

As discussed before, modern automation systems provide large amounts of data and it's seldom wise to transfer all of it to cloud analytics. Therefore in addition to aggregation, the data should be also, at least partially, analyzed locally. The state of the art for this kind of local analytics include training models in a cloud environment with big data. After its performance is sufficient, the model is transferred to the edge and used there with local data. This should also be a continuous process and the model should be trained further and deployed regularly to the edge. This kind of system allows taking full advantage of all the data produced with small delays and offers savings as data is not ingested to a consumption based platform.

Both AWS, Azure and Predix all offer their own edge solution that have very similar features. They all act as cloud gateways for devices, offer device management and security features, allow running analytics and Docker containers. Azures IoT Edge also works as a local MQTT broker which makes it compatible with OPC UA PubSub standard.

Siemens states on their website that they have a "The Siemens Industrial Edge" platform, which includes also all of the above, but does not provide any insights or details for it. It was only released in April 2018 so taking into account the lack of information provided leads to questioning the maturity of it. [131]

Kepserver offers some edge-like functionalities like data conditioning and reduction, but no real edge computing that would allow doing analytics or smart decisions on the edge. Also Thingworx relies on Kepserver when collecting data, and does not provide any information on additional edge computing solutions. Therefore the edge capabilities of Thingworx should be investigated further.

All edge computing solutions discussed provide at least some level of buffering for connectivity disruptions. They range from simple buffers like in Siemens's Mindsphere connectivity boxes to AWS's and Azure's device shadows, which will update local models of entities and synchronize them when connectivity is restored.

6.4 Device Management

All the solutions covered in this thesis were investigated regarding features for managing devices because both the customer and Solita's expert highlighted that it is an important theme when the device count starts to rise. All solutions evaluated had some kind of an answer to the problem and provided all very similar features: mass provisioning, OTA software updates, device monitoring and remote management. AWS was the only vendor that stood up in this theme with their AWS IoT Device Defender, which provided security scanning and auditing to the IoT-network.

However the inspection done was very lightweight and did not benchmark the solutions with detail. Very probably vendors specialized in serving industrial customers have their solutions tailored to that kind of usage, where as open cloud providers need to provide a solution that can be customized to almost every use case. Altogether this thesis cannot find in this scope any significant differences between the vendors regarding device management.

6.5 Integration to cloud

Integration to the cloud covers the connectivity between the factory (edge) and the chosen cloud service. As network connections can never be trusted 100%, the connectivity needs to be supervised and maintained by a system. Usually this is handled by a edge gateway mentioned before, but it can also be done with a simpler gateway.

6.5.1 Physical Connectivity

The factory in the scope of this thesis has exceptionally good Internet connection, as it's located near a big city. This allows a fiber optic connection with a decent 4G backup link to be used for cloud communication. Inside the factory, the office and factory networks are only separated by a firewall, which means devices on the factory floor can get Internet connectivity with just with changing the firewall rules. However the level of connectivity redundancy should be evaluated in the same scope as the rest of the solution, as different solutions require different levels of redundancy: if our factory cannot survive couple hours without the Internet, installing a second fiber optic connection from an another operator running a different physical route would be smart.

6.5.2 Protocol

Choosing the right protocol is highly dependent on how we transfer our data to the cloud. The two option are streaming the data or moving it in batches. Batch transfer is often easier, specially if the source system is old and has no means of producing a adequate data stream for our needs. With batches the most used protocols are File Transfer Protocol (FTP), SSH File Transfer Protocol (SFTP) and HTTP. However using a batch transfer for moving our

data causes delays and makes real time analytics much worse or impossible. Therefore a modern analytics application should favor streaming data always when possible.

For streaming IoT data two most prominent open protocols have emerged: AMQP and MQTT. These both protocols are also supported by majority of the products and platforms represented in this thesis (AWS IoT does not however support AMQP at the time of writing). MQTT is a protocol designed for low overhead and M2M/IoT-communication in mind, where as AMQP is a bit more recent and offers much more functionality like queuing etc. However both of them are very lightweight when compared to traditional web protocols like HTTP. Both protocols also support very similar security features, as they are both based on TCP/IP. Assuming that proper certificates are installed in a secure way and handled as they should, both protocols offer similar security.

One theoretical option would be also to expose the OPC-UA layer to the cloud directly, as it supports sufficient security. This would follow better the paradigm of Industry 4.0 as it exposes the machine layer without any additional gateways or systems between. However a direct network connection to the factory network would require all devices to support these security features and also be updated regularly. For this thesis' scope it is not feasible, but should be kept in mind in the future.

6.6 Cloud

The cloud domain is the most complex of all domains as it's hard to come up with exact requirements for it. Also the domain is expanding very rapidly. With IIoT platform products comparison is somewhere easier, as they are more focused on a subset of functionality and are not as massive as Azure or AWS. The downside is however the lack of neutral evaluation and information in general - vast majority of insight available are originating from whitepapers originally created for marketing.

When deciding between open cloud platforms often the deciding factor in the end is price. The offering between different cloud providers is very similar, a good example is Snowflake: originally an AWS-only component that was a real competitive edge for Amazon ended also up in Azure. However the cloud platform bill is not usually the deciding factor when considering the total price tag: a big part of the costs when building a custom platform on an open cloud platform come from the implementation work and continuous development. At that point a mature component from one cloud provider might bring considerable saving compared to a competing component in a beta stage. However evaluating the matureness of a component needs to be evaluated component by component, there are no pre-existing spreadsheets for it.

6.6.1 Product or a Tailored Solution

The biggest decision to make in the cloud domain will be whether to choose an IIoT data platform product or to implement an own platform tailored to your needs.

Product advantages and disadvantages

The most important advantage products offer are a standard implementation that receives usually periodical updates, often including new functionality. A well chosen product will also support the use case with domain knowledge from the provider. This can range from consultation how to implement and use the platform in an efficient way, to the way the functionalities are designed in the product.

Secondly, adopting a ready-made platform should be faster and cheaper than developing a tailored solution, as all core functionalities already exist. The majority of the implementation work consists of integrations in and out of the system. However products rarely can be adopted seamlessly and ether the organization must adopt to the logic of the product, or the product needs to be customized for the organization. This customization can be very expensive depending on the software provider. A second downside for customization is that it can affect updatability as some or all of the original platform updates may not be compatible or break existing customer specific customizations.

One aspect to consider with products and their applications provided is modifiability: if an application is lacking desired features, can it be modified to fulfill the need? If they are not editable they have to be recreated fully, which increases the work required to the level of a open cloud platform. Specially with platforms advertising no-code solutions (like Thingworx) attention should be paid on modifiability. Thingworx' statements like "Utilizing simple user-friendly interfaces, visualizations, and easy-to-use tools, it eliminates the need for developer or user expertise in data modeling, complex mathematics, or machine learning" [94] raises quite strong doubts about that aspect. However it is possible that these are just themes created by their marketing and are not the full truth.

Some of the platform providers have started to bring services available directly from the underlying open cloud platform. This is a positive trend as it provides more possibilities for integrations (like directing data streams from Azure IoT Hub to Thingworx) and gives developers more tools to work with when custom features are needed. However until now these have been just single services representing a fraction of the offering of these platforms.

Also other papers have addressed the issues highlighted in this thesis considering these products. In [80] researchers from Fraunhofer institute state the following: "Many manufacturing companies have noticed this shift to service-orientation and have started to build their own cloud-based platforms. Examples are the Bosch IoT Suite, GE Predix or Siemens Mindsphere. However, most of these platforms are tailored around the products and services offered by the company and lack interoperability with other platform providers". Also in this domain, the core requirements for Industry 4.0 still apply: open standards to allow networks spanning across multiple companies. Realizing this vision requires that platform providers prioritize making truly interoperable products that rely on open and common standards, instead of trying to cement their customer base with proprietary hardware and software.

Open Cloud Platforms' Advantages and Disadvantages

Naturally when starting to build a platform from scratch the major disadvantage is the amount of work required. However, that also allows implementing a platform that is tailored to the organization's needs. Also, as it is easily expandable and editable it may outlive a platform product.

Open cloud platforms also support agile development very well as they can be accessed fairly easily and provide numerous supporting tools for developers, like Azure DevOps and AWS Code Pipeline. Open cloud platforms are also documented quite well, and have a wide user base, which means usually somebody else has already solved the problem before. In addition, their pricing model scales very flexibly alongside the development and does not require special contract negotiations. Majority of the services offered are priced with quite fine-grained tiers or according to consumption, which prevents situations where significantly more capacity has to be bought than what is actually needed.

When considering integration, a open cloud platform is a double-edged sword: on the other hand almost everything is possible, and most of the things have been done, so there are often templates or even complete code libraries assisting in the process. However almost nothing is just plug-and-play. With platforms designed specifically for IoT-data, it is possible to provide functionality that allows integrating different data sources and endpoints with just couple click in the user interface. However this all depends on the product and the case in hand.

6.6.2 Reflection to the Derived Requirements

This section collects the evaluations to direct requirements derived from the customer needs.

Possibility to support optimization in the future with a feedback time of less than a second

A reliable feedback loop with a latency under 1000 ms is most likely not achieved without edge computing, as cloud platforms may always have problems that cause an increased latency (like using a fallback region). However implementing an edge gateway that supports computing for the most time sensitive operations would fulfill this requirement quite easily.

Possibility to handle heterogeneous data like pictures etc.

Handling heterogenious data like pictures etc. requires a product that has a very diverse functionality offering or a custom solution. Analyzing pictures locally will require edge computing and transferring them to cloud should probably not be done with MQTT or similar protocols, as they are not meant for big payloads. This requirement suggests pretty strongly, that a solution with good edge computing capabilities should be implemented.

Easy integrability to different systems

As all the solutions covered are cloud native so integrations to web based systems should be quite simple. More established platforms probably have a better change of having a ready made connector which makes the integration much cheaper and faster. However integrating to on-premise systems may require some firewall configurations.

No data losses when connection problems occur

Data losses can be mitigated quite easily with buffering, the buffer size just needs to be defined. A bigger buffer can sustain a longer connectivity loss, but costs more. However the buffering should be addressed when designing the system: analytics and other components should take into account, that at some times data might be old or not available, and not go haywire. Device shadow's are a good concept to mitigate the problem.

State-of-the-art security

All solutions covered in this thesis offer sufficient security when implemented correctly and audited for mistakes. However with IT systems there are always risks, and it should be analyzed what kind of data is stored in the cloud. Also a good Cloud Governance is a part of a good security, specially because cloud environments differ a lot from traditional IT-resources as many aspects are handled by the service provider instead of the organization's own IT [132, p 23].

Also securing a platform located in cloud environments differs vastly from solutions hosted on the premises: even if the initial implementation would be considered very secure at the time of implementation, over time every software will get penetrated somehow and requires updating. Therefore having a state-of-the-art security in systems connected to the internet is more a continuous process of evaluating the current status and reacting to it when required, than a one time project where everything is inspected and audited.

Possibility for running modern big data analytics

This depends very much on the definition of modern big data analytics. At this date it would mean the possibility of running various kinds of models (R, Python) with huge cloud computing power and having a developer friendly tooling. Open cloud platforms provide probably a better experience for experimenting and most recent innovations, but platform products may have more refined tools for specific domains that can be easier to use.

Platforms adaptation and future outlook

This is probably the hardest requirement to reflect. To give a comprehensive answer regarding platform products would require an excellent expertise within various sectors of

industry, and are therefore quite hard to evaluate regarding adaptation and future outlook. From academia's point of view, platforms that embrace openness should be better options, where as platforms that are trying to bind customers in their own ecosystem even more should be avoided. GE's Predix is a quite well established platform with a wide marketplace and solutions for edge computing. Therefore in light of this thesis it looks the most promising.

7. CONCLUSION

One key founding in this thesis was that the research questions are heavily dependant of each other, as many providers researched offer solutions that span from data collection on the edge to storing, analyzing and acting on the data in the cloud. Using an integrated solution creates a level of dependency to a vendor, but provides complex features like device management with a relatively simple setup.

7.1 Factory Level and Transportation to Cloud

On the factory level a definitive first step should be the widespread implementation of OPC-UA. As the lifespan of automation systems is long, it might take a while, but as it has become an industry standard, its adoption should be a priority. This implementation suggestion also includes the latest addition, OPC UA PubSub, to allow standardized integrations in the future.

Another key theme on the factory level that should be evaluated, is the role of a MES in data acquisition, analyzing and forwarding. Ideally it should use OPC UA like all other systems for accessing the automation system's data. However as the majority of devices in this case probably do not support OPC UA, the connection needs to be done partly over OPC or vendor specific protocols. This can be done by the MES itself or with an intermediary like KepServer. Kepserver would then provide data for the MES and cloud based analytics. This allows for a single source of data and does not rely on the MES to forward all data to the cloud. The negative side on this solution is that it is not compatible with OPC UA PubSub and does not provide any real edge computing possibilities.

These negative sides lead to the next conclusion: implement an edge hub from the beginning. It allows for a comprehensive, cloud based device management, smart connection management and will provide the required edge computing resources when required in the future. When the automation system has reached a level where everything supports OPC UA, Kepserver can be omitted and the system can be fully based on OPC UA standards.

At least Predix, AWS and Azure offer a promising edge solution that can take care of device management and allows implementing complex systems leveraging edge computing. However they are tightly integrated to rest of their offering, which guides strongly in picking one vendor for all of the domain (factory, connectivity and cloud) and may lead to stronger commitment towards a single vendor than desired.

The answers to the research questions ”How should production data be gathered from an Siemens S7 process logic controllers?” and ”How should the gathered data be transferred

to a cloud service?" could be summarized as following: Aim to gather data from PLCs with an edge hub using OPC-UA. Depending on the selected platform and its support for different protocols, choose one that is designed for IoT like MQTT or AMQP and use it to push data to a cloud endpoint.

7.2 Cloud Level

On the cloud level a key outcome is that a project should not start with choosing a single cloud provider, but instead in defining what are the real capabilities and how our visions match the visions of the vendor. After the case and vision is defined, the next question arising is probably "Should we do it ourselves or use a product made for this".

Buying an IIoT data (and analytics) platform product is a risky move. Many vendors have designed their products with the intent to provide an ecosystem for their own products, and their own products only. However this could be changing slowly due the key themes included in Industry 4.0: common open standards. Another risk with platforms is the limited customization: with products there will always be some compromises that need to be made as they have a limited customizability.

Products however provide some big advantages: if the product is already well established, it has probably found solutions to most common problems and the vendor has some domain knowledge and therefore can provide domain specific assistance. In addition a product saves usually a lot of initial implementation costs, as the core already exists and most of the work is customization and integrations.

One key metric for a product platform is their marketplace, as most of the products are fundamentally based on the same open source software. When benchmarking marketplaces, attention should be paid not only to the total count of applications, but their relevancy to the domain, creator, update frequency and support. If private companies have created apps to a marketplace, their business case should be evaluated also: maintaining an app on a dead marketplace with little customers is a bad business and will lead to the end of maintenance.

One interesting feature with platform products is the ability to use the underlying cloud platform (usually Azure or AWS). This gives interesting possibilities to circumvent possible limitations on the platform regarding e.g. some new analytic services. However usually the services exposed to the platform product are quite limited.

Finally the case of taking an exit from a platform. With platform products this can be very expensive: infrastructure created with the platform's native tools cannot usually be exported and therefore have to be created again from scratch. Custom code created for containers etc. can probably be used again. Exporting big amounts of data can become also expensive, depending on the pricing of the specific platform.

With open cloud platforms the price of an exit depends heavily on the architecture. The analytics are usually created mostly with open source technologies and are therefore fully

exportable. The infrastructure usually uses native components (like FaaS, queue handlers, storage services) with their own APIs that will require migrations. Data exports from open cloud platforms are also not free, but probably still less expensive than from platform products.

Altogether, a suggestion could be given that a custom data platform built on top of an open cloud platform is a more versatile solution that can provide a safer future when built and operated properly. However as more control is given to the organization itself, they also have more responsibility. A platform product on the other hand will restrict some choices and tie the organization to the platform provider, but when chosen well will solve many problems in behalf of the customer organization. They will also somewhat guide the customers vision with the features available versus a custom platform where there needs to be a vision before functionality can be implemented.

7.3 Cultural aspects

During the writing of this thesis, a theme that surfaced very often was the cultural and social requirements for a successful implementation of the Industry 4.0 principles. For management and business owners this means among other thing understanding how to use data for creating value, creating visions about ”What does all this mean for our company” and committing to the actions created to achieve those visions.

For regular workers the requirements are a bit similar, as less workforce is required on the shop floor, the demand for leadership and technical roles increase. This means that more education is required and as the development speed of things increase, also the demand for re-education and lifelong learning increases.

To sum this up, a successful implementation for providing machine level data for cloud based analytics is a very interwoven topic. It requires competent personal with a clear vision, it requires the right mix of technology, partners and vendors that depend case by case and it will not be a project that ends completely at some point.

BIBLIOGRAPHY

[1] *A Conversation with Ginni Rometty*. Youtube, website. URL: https://www.youtube.com/watch?v=SUoCHC-i7_o (visited on 02/02/2019).

[2] K. Zhou, T. Liu, and L. Zhou. "Industry 4.0: Towards future industrial opportunities and challenges". English. In: IEEE, 2015, pp. 2147–2152.

[3] J. Schlechtendahl, M. Keinert, F. Kretschmer, A. Lechler, and A. Verl. "Making existing production systems Industry 4.0-ready". In: *Production Engineering* 9.1 (Feb. 2015), pp. 143–148. ISSN: 1863-7353. DOI: 10.1007/s11740-014-0586-3. URL: https://doi.org/10.1007/s11740-014-0586-3.

[4] L. Monostori, B. Kádár, T. Bauernhansl, S. Kondoh, S. Kumara, G. Reinhart, O. Sauer, G. Schuh, W. Sihn, and K. Ueda. "Cyber-physical systems in manufacturing". English. In: *CIRP Annals - Manufacturing Technology* 65.2 (2016), pp. 621–641.

[5] A. Gilchrist. *Industry 4.0*. English. DE: Apress, 2016. ISBN: 1484220463.

[6] *Gartner IT Glossary: Analytics*. English. Gartner, website. URL: https://www.gartner.com/it-glossary/analytics/ (visited on 01/09/2019).

[7] *Gartner IT Glossary: Advanced Analytics*. English. Gartner, website. URL: https://www.gartner.com/it-glossary/advanced-analytics/ (visited on 01/09/2019).

[8] Frost&Sullivan. *Advanced Analytics: Disruptive Opportunities*. English. Tech. rep. Frost & Sullivan, 2017. URL: https://cds-frost-com.libproxy.tut.fi/p/56579/#!/ppt/c?id=D7DA-01-00-00-00.

[9] C. Verdouw, R. Robbemond, and J. W. Kruize. "Integration of Production Control and Enterprise Management Systems in Horticulture". In: Sept. 2015.

[10] J. Kletti and I. Books24x7. *Manufacturing Execution Systems — MES*. English. DE: Springer Verlag, 2007. ISBN: 9783540497431.

[11] Frost&Sullivan. *Global Data Analytics for Industries Report*. English. Tech. rep. Frost&Sullivan, 2017. URL: https://cds-frost-com.libproxy.tut.fi/p/56579/#!/ppt/c?id=K096-01-00-00-00.

[12] P. Lade, R. Ghosh, and S. Srinivasan. "Manufacturing Analytics and Industrial Internet of Things". English. In: *IEEE Intelligent Systems* 32.3 (2017), pp. 74–79.

[13] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu. "Edge Computing: Vision and Challenges". English. In: *IEEE Internet of Things Journal* 3.5 (2016), pp. 637–646.

[14] *Hype Cycle for Cloud Computing, 2018*. English. Gartner, website. URL: https://www.gartner.com/doc/3884671/hype-cycle-cloud-computing- (visited on 01/09/2019).

[15] L. C. Mitch Tseng Todd Edmunds. *Introduction to Edge Computing*. English. Tech. rep. Industrial Internet Consortium, 2017. URL: https://hub.iiconsortium.org/portal/Whitepapers/5bbdbada91337a000f7b1b2f.

[16] D. J. H. Prof. Dr. Henning Kagermann Prof. Dr. Wolfgang Wahlster. *Recommendations for implementing the strategic initiative INDUSTRIE 4.0*. English. Tech. rep. German National Academy of Science and Engineering, 2013. URL: https:

//www.acatech.de/wp-content/uploads/2018/03/Final_report__Industrie_4.0_accessible.pdf.

[17] N. Jazdi. "Cyber physical systems in the context of Industry 4.0". In: *2014 IEEE International Conference on Automation, Quality and Testing, Robotics*. May 2014, pp. 1–4. DOI: 10.1109/AQTR.2014.6857843.

[18] E. A. Lee and S. A. Seshia. *Introduction to Embedded Systems - A Cyber-Physical Systems Approach, Second Edition*. English. MIT Press, 2011. ISBN: 978-0-262-53381-2.

[19] B. Vogel-Heuser, J. Lee, and P. Leitão. "Agents enabling cyber-physical production systems". English. In: *at - Automatisierungstechnik* 63.10 (2015), pp. 777–789.

[20] L. Monostori. "Cyber-physical Production Systems: Roots, Expectations and R&D Challenges". In: *Procedia CIRP* 17 (2014). Variety Management in Manufacturing, pp. 9–13. ISSN: 2212-8271. DOI: https://doi.org/10.1016/j.procir.2014.03.115. URL: http://www.sciencedirect.com/science/article/pii/S2212827114003497.

[21] Y. Chen. "Integrated and Intelligent Manufacturing Perspectives and Enablers". Chinese. In: 3.5 (2017), pp. 588–595.

[22] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin. "Smart Factory of Industry 4.0: Key Technologies, Application Case, and Challenges". English. In: *IEEE Access* 6 (2018), pp. 6505–6519.

[23] *Azure SQL Data Warehouse - Massively parallel processing (MPP) architecture*. Microsoft, website. 2018. URL: https://docs.microsoft.com/en-us/azure/sql-data-warehouse/massively-parallel-processing-mpp-architecture (visited on 09/01/2018).

[24] *What is a data lake?* AWS, website. 2018. URL: https://aws.amazon.com/big-data/datalakes-and-analytics/what-is-a-data-lake/ (visited on 09/01/2018).

[25] *What is a Reference Architecture?* HP, website. 2018. URL: https://www.hpe.com/us/en/what-is/reference-architecture.html (visited on 09/01/2018).

[26] *Agile Data*. Solita, website. 2018. URL: https://www.solita.fi/agile-data/ (visited on 09/01/2018).

[27] Frost&Sullivan. *Global IoT Platforms Trends, 2017*. English. Tech. rep. Frost&Sullivan, 2017. URL: https://cds-frost-com.libproxy.tuni.fi/p/298946756/#!/ppt/c?id=MD14-01-00-00-00&hq.

[28] N. Naik. "Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP". In: *2017 IEEE International Systems Engineering Symposium (ISSE)*. Oct. 2017, pp. 1–7. DOI: 10.1109/SysEng.2017.8088251.

[29] T. Yokotani and Y. Sasaki. "Comparison with HTTP and MQTT on required network resources for IoT". English. In: IEEE, 2016, pp. 1–6.

[30] *Hypertext Transfer Protocol – HTTP/1.1*. Tech. rep. Internet Engineering Task Force, 1999. URL: https://tools.ietf.org/html/rfc2616#section-1.

[31] *Evolution of HTTP*. Mozilla, website. 2018. URL: https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/Evolution_of_HTTP (visited on 01/17/2019).

[32] *What is SSL, TLS and HTTPS?* Symantec, website. URL: https://www.websecurity.symantec.com/security-topics/what-is-ssl-tls-https (visited on 01/17/2019).

[33] *MQTT - Frequently Asked Questions*. MQTT, website. URL: http://mqtt.org/faq (visited on 02/02/2019).

[34] *MQTT Security Fundamentals: TLS/SSL*. HiveMQ , website. URL: https://www.hivemq.com/blog/mqtt-security-fundamentals-tls-ssl/ (visited on 02/02/2019).

[35] R. Cohn. *A Comparison of AMQP and MQTT*. English. Tech. rep. StormMQ, 2017. URL: https://lists.oasis-open.org/archives/amqp/201202/msg00086/StormMQ_WhitePaper_-_A_Comparison_of_AMQP_and_MQTT.pdf.

[36] *Aws IoT Protocols*. AWS, website. 2019. URL: https://docs.aws.amazon.com/iot/latest/developerguide/protocols.html (visited on 04/01/2019).

[37] *What is OPC?* OPC Foundation , website. URL: https://opcfoundation.org/about/what-is-opc/ (visited on 02/02/2019).

[38] *What is an OPC Server and an OPC Client?* Matrikon , website. URL: https://www.matrikonopc.com/resources/opc-server.aspx/ (visited on 02/02/2019).

[39] *OPC ja OPC UA*. Novotek , website. URL: https://www.novotek.com/fi/ratkaisut/kepware-kommunikointialusta/opc-ja-opc-ua (visited on 02/02/2019).

[40] *Unified Architecture*. OPC Foundation , website. URL: https://opcfoundation.org/about/opc-technologies/opc-ua/ (visited on 02/02/2019).

[41] *OPC UA*. Siemens , website. URL: https://www.siemens.com/global/en/home/products/automation/industrial-communication/opc-ua.html (visited on 02/02/2019).

[42] *OPC Unified Architecture Specification Part 14: PubSub*. Tech. rep. 1.04. OPC Foundation, 2018.

[43] *Eclipse Mosquitto™ An open source MQTT broker*. Mosquitto , website. URL: https://mosquitto.org/ (visited on 02/02/2019).

[44] *OPC Foundation MQTT Prototyper*. OPCFoundation, Github. URL: https://github.com/OPCFoundation/UA-.NETStandard/tree/prototyping-mqtt/Mqtt (visited on 02/02/2019).

[45] *Should I Use OPC UA or MQTT or AMQP?* OPCFoundation, website. URL: https://opcconnect.opcfoundation.org/2017/10/should-i-use-opc-ua-mqtt-amqp/ (visited on 02/02/2019).

[46] *Siemens Annual Repost 2017*. Siemens, pdf. URL: https://www.siemens.com/investor/pool/en/investor_relations/Siemens_AR2017.pdf (visited on 02/02/2019).

[47] *Industrial Automation Systems SIMATIC*. Siemens, website. URL: https://www.siemens.com/global/en/home/products/automation/systems/industrial.html (visited on 02/02/2019).

[48] *PROFINET*. Siemens, website. URL: https://www.siemens.com/global/en/home/products/automation/industrial-communication/profinet.html (visited on 02/02/2019).

[49] *SIMATIC OPC UA S7-1500*. Siemens, website. URL: https://w3.siemens.com/mcms/automation-software/en/tia-portal-software/step7-tia-portal/simatic-step7-options/opc-ua-s7-1500/pages/default.aspx (visited on 02/02/2019).

[50] *About Kepware*. Kepware, website. URL: https://www.kepware.com/en-us/about/overview/ (visited on 02/02/2019).

[51] *Kepware IoT Gateway*. Kepware, website. URL: https://www.kepware.com/en-us/products/kepserverex/advanced-plug-ins/iot-gateway/ (visited on 02/02/2019).

[52] *KEPServerEX*. Kepware, website. URL: https://www.kepware.com/en-us/products/kepserverex/ (visited on 02/02/2019).

[53] *Kepserver Configuration API*. Kepserver, website. URL: https://www.kepware.com/en-us/products/kepserverex/features/configuration-api/ (visited on 02/12/2019).

[54] *AWS Greengrass*. English. AWS, website. 2018. URL: https://aws.amazon.com/greengrass/ (visited on 10/25/2018).

[55] *Cloud Market Q3 Snapshot: Azure Is Fastest, But AWS Is Biggest*. AWSinsider, website. 2019. URL: https://awsinsider.net/articles/2018/11/01/azure-fastest-but-aws-biggest.aspx (visited on 04/01/2019).

[56] *AWS Greengrass FAQs*. English. AWS, website. 2018. URL: https://aws.amazon.com/greengrass/faqs/ (visited on 10/25/2018).

[57] *Use Greengrass OPC-UA to Communicate with Industrial Equipment*. English. AWS, website. 2018. URL: https://docs.aws.amazon.com/greengrass/latest/developerguide/opcua.html (visited on 10/25/2018).

[58] *IoT & Industry 4.0 Web Day | AWS Greengrass and OPC-UA Support*. AWS, Youtube. URL: https://www.youtube.com/watch?v=KzUUspC6Tbg (visited on 01/30/2019).

[59] *AWS Partner Device Catalog*. AWS, website. URL: https://devices.amazonaws.com/search?page=1&sv=gg&type=edgeserver (visited on 01/30/2019).

[60] *The NIST Definition of Cloud Computing*. English. Tech. rep. 800-145. Gaithersburg, MD 20899-8930: National Institute of Standards and Technology, 2011. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf.

[61] B. Varghese and R. Buyya. "Next generation cloud computing: New trends and research directions". In: *Future Generation Computer Systems* 79 (2018), pp. 849–861. ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2017.09.020. URL: http://www.sciencedirect.com/science/article/pii/S0167739X17302224.

[62] *Amazon S3 Pricing*. AWS, website. 2018. URL: https://aws.amazon.com/s3/pricing/ (visited on 09/01/2018).

[63] E. van Eyk, L. Toader, S. Talluri, L. Versluis, A. Uță, and A. Iosup. "Serverless is More: From PaaS to Present Cloud Computing". In: *IEEE Internet Computing* 22.5 (Sept. 2018), pp. 8–17. ISSN: 1089-7801. DOI: 10.1109/MIC.2018.053681358.

[64] *Serverless - The serverless application framework*. Serverless, website. 2018. URL: https://serverless.com/ (visited on 09/01/2018).

[65] *Snowflake Data Warehouse*. Unified Automation, website. 2018. URL: https://www.snowflake.com/product/ (visited on 11/20/2018).

[66] *Snowflake announces general availability on Microsoft Azure*. Unified Automation, website. 2018. URL: https://www.snowflake.com/news/snowflake-announces-general-availability-on-microsoft-azure// (visited on 11/20/2018).

[67] *Cloud Foundry Members*. Cloud Foundry, website. URL: https://www.cloudfoundry.org/members/ (visited on 02/02/2019).

[68] *Cloud Foundry Platform*. Cloud Foundry, website. URL: https://www.cloudfoundry.org/why-cloud-foundry/ (visited on 02/02/2019).

[69] *Try Cloud Foundry*. Cloud Foundry, website. URL: https://www.cloudfoundry.org/how-to-try-cloud-foundry/ (visited on 02/12/2019).

[70] *Pivotal Cloud Foundry (PCF)*. Pivotal, website. URL: https://pivotal.io/platform (visited on 01/30/2019).

[71] *Pivotal Function Service (PFS)*. Pivotal, website. URL: https://pivotal.io/platform/pivotal-function-service (visited on 01/30/2019).

[72] *Pivotal Services Marketplace*. Pivotal, website. URL: https://pivotal.io/platform/services-marketplace (visited on 01/30/2019).

[73] *This is MindSphere!* Siemens, website. URL: https://www.siemens.com/global/en/home/products/software/mindsphere.html (visited on 01/30/2019).

[74] *Siemens Driving Digital Transformation with $10 Billion Investment in U.S. Software Companies since 2007*. Businesswire, website. URL: https://www.businesswire.com/news/home/20170327005329/en/Siemens-Driving-Digital-Transformation-10-Billion-Investment (visited on 01/30/2019).

[75] *Mindsphere applications*. Siemens, website. URL: https://www.dex.siemens.com/mindsphere/Applications (visited on 01/30/2019).

[76] *MindSphereMindConnect IoT Extension*. Siemens, website. URL: https://documentation.mindsphere.io/resources/html/mindconnect-iot-extension/en-US/110459619211.html (visited on 02/12/2019).

[77] *MindAccess*. Siemens, website. URL: https://www.plm.automation.siemens.com/store/en-se/mindsphere/mindaccess/iot.html (visited on 11/15/2018).

[78] *MindSphere Seller Guide*. Siemens, website. URL: https://siemens.mindsphere.io/content/dam/mindsphere/pdf/MindSphere_SellerGuide_v1801.pdf (visited on 11/15/2018).

[79] *What MindConnect elements are available to connect to MindSphere?* Siemens, website. URL: https://community.plm.automation.siemens.com/t5/MindSphere-FAQs/What-MindConnect-elements-are-available-to-connect-to-MindSphere/tap/424753 (visited on 01/30/2019).

[80] B. Götz, D. Schel, D. Bauer, C. Henkel, P. Einberger, and T. Bauernhansl. "Challenges of Production Microservices". English. In: *Procedia CIRP* 67 (2018), pp. 167–172.

[81] *CEO letter 2017*. GE, website. 2017. URL: https://www.ge.com/investor-relations/ar2017/ceo-letter (visited on 11/20/2018).

[82] *Predix platform*. GE, website. 2018. URL: https://www.ge.com/digital/predix-platform-foundation-digital-industrial-applications (visited on 11/20/2018).

[83] *GE Leverages Pivotal Cloud Foundry to Build Predix, First Cloud for Industry*. Cloud Foundry, website. URL: https://www.cloudfoundry.org/blog/ge-leverages-pivotal-cloud-foundry-to-build-predix/ (visited on 02/12/2019).

[84] *Predix Services*. GE, website. URL: https://www.predix.io/catalog/services/ (visited on 01/30/2019).

[85] *Global Partner Summit 2017: Competing to Win - Predix Pricing and Use Cases*. GE, pdf. URL: https://www.ge.com/digital/sites/default/files/Global-Partner-Summit-2017-Predix-Pricing-Use-Cases-Justin-LaChance-Jason-Seay.pdf (visited on 01/30/2019).

[86] *Predix Free Tier FAQ*. GE, website. 2016. URL: https://forum.predix.io/articles/14550/predix-free-tier-faq.html (visited on 09/01/2018).

[87] Frost&Sullivan. *Intelligence at the Edge—An Outlook on Edge Computing*. English. Tech. rep. Frost & SUllivan, 2017. URL: https://cds-frost-com.libproxy.tut.fi/p/56579/#!/ppt/c?id=K198-01-00-00-00.

[88] *Predix Edge from GE Digita*. GE, pdf. 2018. URL: https://www.ge.com/digital/sites/default/files/download_assets/predix-edge-ge-digital-datasheet.pdf (visited on 09/01/2018).

[89] *Predix Edge Manager*. GE, website. 2018. URL: https://www.predix.io/services/service.html?id=1583 (visited on 09/01/2018).

[90] *Hydraulivasaroiden valmistus tehostuu IoT:n avulla*. Finnish. Elisa Oyj, website. 2017. URL: https://hub.elisa.fi/hydraulivasaroiden-valmistus-tehostuu-iotn-avulla/ (visited on 10/25/2018).

[91] PTC. *Deployment Architecture Guide*. English. PTC. 2018. URL: https://support.ptc.com/WCMS/files/174041/en/Thingworx_Deployment_Architecture_Guide.pdf.

[92] *PTC ThingWorx Picks Microsoft's Azure as Preferred IIoT Cloud*. English. IoT World Today, website. 2018. URL: https://www.iotworldtoday.com/2018/02/16/ptc-thingworx-picks-microsofts-azure-preferred-iiot-cloud/ (visited on 10/25/2018).

[93] *Why Marketplaces Are Key for Industrial IoT Platforms*. English. PTC, website. 2018. URL: https://www.ptc.com/en/thingworx-blog/why-marketplaces-are-key-for-industrial-iot-platforms (visited on 10/25/2018).

[94] PTC. *ThingWorx Platform - Product Brief*. English. Tech. rep. PTC, 2018. URL: https://www.ptc.com/-/media/Files/PDFs/ThingWorx/ThingWorx_Platform-Product-Brief_2018.pdf.

[95] *Which ThingWorx Trial is right for you?* Thingworx, website. 2018. URL: https://developer.thingworx.com/resources/downloads (visited on 09/01/2018).

[96] *ThingWorx 8 Utilities*. Thingworx, website. URL: https://support.ptc.com/help/thingworx_hc/thingworx_utilities_8_hc (visited on 02/12/2019).

[97] *ThingWorx Asset Advisor*. Thingworx, website. URL: https://www.ptc.com/en/products/service-lifecycle-management/thingworx-asset-advisor (visited on 02/12/2019).

[98] *PTC Marketplace*. Thingworx, website. 2018. URL: https://marketplace.ptc.com/home (visited on 09/01/2018).

[99] *Welcome to ThingWorx Edge SDKs and WebSocket-based Edge MicroServer (WS EMS) Help Center*. Thingworx, website. 2018. URL: http://support.ptc.com/help/thingworx_hc/thingworx_edge_sdks_ems/#page/latest (visited on 09/01/2018).

[100] *Gartner Says Worldwide IaaS Public Cloud Services Market Grew 29.5 Percent in 2017*. Gartner, website. 2018. URL: https://www.gartner.com/newsroom/id/3884500 (visited on 09/01/2018).

[101] *Google avasi "paikallisen pilven" Haminaan – Voisiko valtio nyt siirtää miljoonien suomalaisten tiedot Googlelle?* Helsingin sanomat, website. 2018. URL: https://www.hs.fi/teknologia/art-2000005818443.html (visited on 09/01/2018).

[102] S. Somashekhar, R. Kumarc, A. Rauthan, K. Arun, P. Patil, and Y. Ramya. "Abstract S6-07: Double blinded validation study to assess performance of IBM artificial intelligence platform, Watson for oncology in comparison with Manipal multidisciplinary tumour board – First study of 638 breast cancer cases". English. In: *Cancer Research* 77.4 Supplement (2017).

[103] *Roundup Of Machine Learning Forecasts And Market Estimates, 2018*. Forbes, website. 2018. URL: https://www.forbes.com/sites/louiscolumbus/2018/02/18/roundup-of-machine-learning-forecasts-and-market-estimates-2018/#76c584722225 (visited on 09/01/2018).

[104] *10 years*. AWS, website. 2016. URL: https://aws.amazon.com/10year/ (visited on 09/01/2018).

[105] *Cloud Products*. AWS, website. 2018. URL: https://aws.amazon.com/products/?nc2=h_ql_prod (visited on 09/01/2018).

[106] *Case Studies*. AWS, website. 2018. URL: https://aws.amazon.com/solutions/case-studies/ (visited on 09/01/2018).

[107] *Aws Free Tier*. AWS, website. 2018. URL: https://aws.amazon.com/free/ (visited on 09/01/2018).

[108] *Aws Lambda*. AWS, website. 2018. URL: https://aws.amazon.com/lambda/ (visited on 09/01/2018).

[109] *Aws IoT Core*. AWS, website. 2018. URL: https://aws.amazon.com/iot-core/ (visited on 09/01/2018).

[110] *Aws IoT Core Features*. AWS, website. 2018. URL: https://aws.amazon.com/iot-core/features/ (visited on 09/01/2018).

[111] *AWS IoT Device Management Features*. AWS, website. URL: https://aws.amazon.com/iot-device-management/features/ (visited on 02/12/2019).

[112] *AWS IoT Device Defender*. AWS, website. URL: https://aws.amazon.com/iot-device-defender/ (visited on 02/12/2019).

[113] *Amazon SageMaker Instance Types*. AWS, website. 2018. URL: https://aws.amazon.com/sagemaker/pricing/instance-types/ (visited on 09/01/2018).

[114] *SageMaker vs. DLAMI*. AWS, forums. 2018. URL: https://forums.aws.amazon.com/thread.jspa?messageID=828584&tstart=0 (visited on 09/01/2018).

[115] *Amazon SageMaker*. AWS, website. 2018. URL: https://aws.amazon.com/sagemaker/ (visited on 09/01/2018).

[116] *What is Azure?* Microsoft, website. 2018. URL: https://azure.microsoft.com/en-us/overview/what-is-azure/ (visited on 09/01/2018).

[117] *Microsoft launches Windows Azure*. Cnet, website. 2008. URL: https://www.cnet.com/news/microsoft-launches-windows-azure/ (visited on 09/01/2018).

[118] *What is Azure IoT Hub?* Microsoft, website. 2018. URL: https://docs.microsoft.com/en-us/azure/iot-hub/about-iot-hub (visited on 09/01/2018).

[119] *Communicate with your IoT hub using the MQTT protocol*. Microsoft, website. 2018. URL: https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-mqtt-support (visited on 09/01/2018).

[120] *What is Azure IoT Edge*. Microsoft, website. 2018. URL: https://docs.microsoft.com/en-us/azure/iot-edge/about-iot-edge (visited on 09/01/2018).

[121] *Understand the Azure IoT Edge runtime and its architecture*. Microsoft, website. 2018. URL: https://docs.microsoft.com/en-us/azure/iot-edge/iot-edge-runtime (visited on 09/01/2018).

[122] *Azure Machine Learning Service Documentation*. Microsoft, website. 2018. URL: https://docs.microsoft.com/en-us/azure/machine-learning/service/ (visited on 09/01/2018).

[123] *What is Azure Stream Analytics?* Microsoft, website. 2018. URL: https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-introduction (visited on 09/01/2018).

[124] *What is Azure Analysis Services?* Microsoft, website. 2018. URL: https://docs.microsoft.com/en-us/azure/analysis-services/analysis-services-overview (visited on 09/01/2018).

[125] *Azure Data Lake Storage*. Microsoft, website. 2018. URL: https://azure.microsoft.com/en-us/services/storage/data-lake-storage/ (visited on 09/01/2018).

[126] *Azure Cosmos DB*. Microsoft, website. 2018. URL: https://azure.microsoft.com/en-us/services/cosmos-db/ (visited on 09/01/2018).

[127] *What is Azure Stack?* Microsoft, website. 2018. URL: https://azure.microsoft.com/en-gb/overview/azure-stack/ (visited on 09/01/2018).

[128] R. Hepworth. *Azure Stack: Lessons learned from use in the real world*. 2018. URL: https://github.com/rikhepworth/Presentations/raw/master/PolarConf%20Azure%20Stack%20Lessons%20Learned%20From%20The%20Real%20World.pptx (visited on 09/01/2018).

[129] *Siemens industrial communication relies on Time-Sensitive Networking (TSN)*. Siemens, website. 2018. URL: https://www.siemens.com/press/en/pressrelease/?press=/en/pressrelease/2018/processindustries-drives/pr2018040189pden.htm&content[]=PDr (visited on 11/20/2018).

[130] *OPC UA Publish-Subscribe (Pub/Sub) - IoT becomes easier*. Unified Automation, website. 2018. URL: https://www.unified-automation.com/news/news-details/article/opc-ua-publish-subscribe-pubsub-iot-becomes-easier.html (visited on 11/20/2018).

[131] *Industrial Edge from Siemens adds benefits from the cloud at the field level*. Siemens, press release. URL: https://www.siemens.com/press/en/pressrelease/?press=/en/pressrelease/2018/digitalfactory/pr2018040239dfen.htm (visited on 02/02/2019).

[132] S. Blount and R. Zanella. *Cloud Security and Governance : Who's on Your Cloud?*. IT Governance Publishing, 2010. ISBN: 9781849280907. URL: http://search.ebscohost.com/login.aspx?direct=true&AuthType=cookie,ip,uid&db=nlebk&AN=391120&site=ehost-live&scope=site&authtype=sso&custid=s4778523.

APPENDIX A: INTERVIEW WITH A SOLITA EXPERT

Interview with a Solita Data Scientist on 25.7.2018 at 9 am

Question 1: Who are you and what do you do?

Olen titteliltäni Data Engineer Solita Oy:ssä. Teen paljon AWS- ja pilvipalveluprojekteja. Kuulun Solitan IoT-henkiseen datatiimiin ja teemme pääasiassa IoT-projekteja. Näissä projekteissa vastuualueemme on yleensä enemmän datan siirron, pilven ja data-analytiikan puolella, itse datan keruu sensoreilla on yleensä yhteistyökumppaniemme tai asiakkaamme vastuulla.

Question 2: How does the pricing between products (like GE's Predix) differ from cloud platform like AWS?

Minulla ei ole omakohtaista kokemusta tuotteista kuten Predix, joskin eräs asiakkaamme harkitsi sitä omaksi IoT-alustakseen, kuitenkin päätyen lopulta AWS:ään. Pilvipalveluiden (kuten AWS) etuna on erittäin hyvä hinnoittelun skaalautuminen, hinnoittelu määräytyy hyvin tarkasti oman käytön mukaan, oli käytössä sitten 10 tai 100 000 laitetta.

Arvioisin että valmiiden tuotteiden haittapuolina voi olla voimakkaampi alustaan ja teknologioihin sitoutuminen. Avoimissa pilvialustoissa näitä rajoitteita on vähemmän. Tuotepuoli yhdistyy vahvasti pilvipalveluihin, kuten esim. Siemens Mindsphere on rakennettu AWS:n päälle, ja siten tarjoaa mahdollisuuden sen käyttöön suoraan asiakkaan AWS-pilvestä.

Pilvialustojen hintojen vertailu on haastavaa, koska eri palvelut käyttävät eri parametreja hinnoittelussaan, esimerkiksi toinen palvelu voi veloittaa käyttöminuuttien mukaan, toinen liikennemäärän mukaan. Ainakin AWS, oletettavasti myös muut palveluntarjoajat, kehittävät myös aktiivisesti hinnoitteluaan ja pyrkivät tekemään siitä entistä joustavampaa, jolloin hinnoittelu määräytyy tarkemmin käytettyjen resurssien mukaan, jolloin asioita niputetaan vähemmän yhteen.

Question 3: According to your knowledge and experience, how would you compare Google's, Azure's and AWS's IoT services?

Googlen alustasta minulla ei ole kokemusta, joten vastaan Azuren ja AWS:n osalta. Tuettujen protokollien osalta Azurella on hienoinen etu, mutta molemmista löytyvät oikeastaan kaikki tarvittavat, esimerkiksi MQTT. Molemmat tarjoavat laajalla skaalalla tukevia toiminnallisuuksia itse ”perus” IoT-toiminnallisuuksien ympärille, kuten laitehallintaa,

analytiikkaa yms. ja rakentavat siten omaa ekosysteemiänsä. AWS osalta näitä ovat mm. Greengrass (gateway), FreeRTOS(mikrokontrolleri käyttöjärjestelmä) ja IoT Analytics.

Molemmissa palveluissa on tietoturvaa painotettu vahvasti. Ainakin AWS:ssä on herätty hieman jälkijunassa laitehallinnan tärkeyteen, tähän asti työkaluja suurien laitemäärien käsittelyyn ei juuri ole ollut, mutta viime aikoina julkaistu Device Management-sovellus todennäköisesti ratkoo tätä ongelmaa.

IoT-sovellukselle alustaa valittaessa ei useinkaan olla tyhjiössä, vaan IoT-laitteet ovat vain yksi datan lähde muun joukossa. Tällöin valinnassa monesti painotetaan enemmän analytiikkaominaisuuksia tai suositaan jo organisaation käytössä olevaa pilvialustaa.

Erityisesti AWS:n kohdalla olen huomannut, että he tuovat markkinoille hyvin aggressiivisesti uusia versioita tuotteista. Nämä versiot ovat kylläkin toimivia, mutta usein hyvin raakileita. Mikäli nämä raakileet saavat suosiota käyttäjien parissa, niitä kuitenkin kehitetään hyvin aktiivisesti ja nopeasti. Ainakin AWS IoT käyttäjiä on tällä hetkellä runsaasti, jonka vuoksi sen kehityksen tunnutaankin panostavan aika hyvin.

Question 4: What kind of special characteristics does an industrial IoT application have when compared to applications aimed for consumer usage?

Molemmissa osa-alueissa on paljon yhteisiä piirteitä, kuten tietoturvan tärkeys. Eri sovelluskohteissa on tietysti eroja, esimerkiksi kodin lämpötilan vuotaminen ulkopuoliselle ei yleensä ole katastrofi, mutta liikesalaisuudeksi luokiteltavan prosessin lämpötilat eivät missään nimessä saa vuotaa ulkopuolisille.

Teollisuuspuolella sovellukset ja niiden toimintaympäristöt ovat usein helpommin määriteltävissä, joskin usein haastavimpia esimerkiksi lämpötilan, kosteuden ja värinän suhteen. Lisäksi teollisuuspuolella vaatimukset ovat usein kovempia kuin kuluttajasovelluksissa. Tämän seurauksena myös itsediagnostiikan tärkeys korostuu. Esimerkiksi AWS:n ekosysteemissä on esimerkiksi poikkeuksien löytämiseen suunnattu Anomaly detection, joka pyrkii löytämään poikkeuksellisia tapahtumia suurista datamassoista. Teollisuuspuolella IoT-sovellukset tuntuvat myös usein rakentuvat automaatiojärjestelmien yhteyteen.

Question 5: Do you have any experience on working and cleaning data, which is originated from a PLC system?

En ole itse työstänyt PLC:stä tullutta dataa, mutta yleisesti voisi sanoa, että kaikkea data pitää jonkin verran siistiä. Lähtökohtaisesti kuitenkin PLC:stä tuleva data on usein aika luotettavaa ja tarvitsee varmaankin keskimääräistä vähemmän siivoamista.

Question 6: What are your thoughts on AWS Greengrass?

Näen että Greengrass on käytännössä monenlaisia palveluita tarjoava gateway, joka osaa itsekseen mm. hallinnoida konnektiviteettiä – yhteyden katketessa tieto tallennetaan lokaalisti ja sen palautuessa katkon aikana tallennettu data siirretään pilveen automaattisesti.

Samaa ekosysteemiä laajentaa FreeRTOS-käyttöjärjestelmä, joka osaa Greengrassin kautta hankkia itselleen mm. konnektiviteetin ja tarvittavat sertifikaatit. Tämän johdosta jokaista sensoria ei tarvitse konfiguroida erikseen. Lisäksi samassa ekosysteemistä löytyvät anomaly detection, laitehallinta sekä analytiikka, niin edgellä kuin pilvessä.

Greengrass tarjoaakin paikan tehdä edge-analytiikkaa, jonka avulla voidaan saavuttaa nopeampia vasteaikoja, mutta myös kustannussäästöjä, koska pilvipalveluiden yksin hinnoitteluperuste on tiedonsiirron määrä. Lisäksi Greengrass pystyy ajamaan AWS:stä tuttuja Lambda-funktioita, jotka ovat jokaiselle AWS-kehittäjälle tuttuja ja siksi helpottaa ympäristöön tutustumista.

Ei ehkä varsinainen ongelma, mutta yksi kompastuskivi saattaa olla itse laitetaso. Greengrass (ja AWS IoT) itsessään on kuitenkin vain ohjelmakoodia, jota pitää kuitenkin suorittaa jossakin laitteessa. Oikeiden laitteiden valintaan tulee kiinnittää huomioita, ja varmistaa että ne täyttävät kaikki vaatimuksen niin tehon, I/O:n ja suojaluokituksien suhteen. En ole toistaiseksi nähnyt, että olisi saatavilla ns. plug-and-play laitteita Greengrassia varten. En kuitenkaan ole käytännössä toteuttanut Greengrassin pohjautuvaa sovellusta, joten tietoni mahdollisista haasteista ovat rajallisia. Paperilla kuitenkin kaikki vaikuttaa aika hyvältä.

Question 7: How would you describe the state of the Art for data analytics and data visualization?

Tähän kysymykseen ei tietenkään ole yhtä oikeasta vastausta, ja jokaisella on varmasti tähän oma mielipiteensä. Ehkä yksi olennaisin asia on kuitenkin pilvilaskenta, joka mahdollista laskentatehon ostamisen minuuttihinnoittelulla. Aluksi Azure oli pitkään edelläkävijä pilvipohjaisessa analytiikassa, ja tarjosi hyvin helposti käyttöönotettavia algoritmeja ja koneoppimispalveluita. AWS on kuitenkin kirinyt tätä etumatkaa kiinni nopeasti, uusimpana esimerkkinä AWS Sage Maker, joka helpottaa analytiikkamallien käyttöönottoa.

Datan tallennuksen ja prosessoinnin siirtyminen pilveen on myös vaikuttanut osaltaan tulosten esityskerrokseen – koska data ja sen tuottama tieto on jo valmiiksi pilvipalvelussa, ei sen visualisointiakaan oikeastaan kannata enää tehdä lokaalisti. Itse analyysien tekoon käytetään monesti valmiita ohjelmia, kuten SPSS ja R, mutta käytössä on myös paljon avoimen lähdekoodin ohjelmakirjastoja. Analytiikka onkin muuttunut hyvin paljon enemmän ohjelmoinniksi. Esimerkiksi avoimia koneoppimiskirjastoja on tarjolla tuhansia eri ohjelmointikielille.

Datan käsittely on pilvipalveluiden myötä monipuolistunut ja siirtynyt lähemmäksi ohjelmointia graafisten eräajoihin keskittyvien ETL työkalujen vähentyessä. Tämän ohessa myös Big Data -analytiikka ja siihen käyttävät metodit ovat nostaneet uudelleen päätään. Lisäksi tarjolle on tullut täysin uusia työkaluja, kuten tekoälypohjainen analytiikka, jossa sovellukset yleensä toteutetaan ohjelmoimalla.

Yhteenvetona sanoisin, että analytiikan State-of-the-art pyörii pilven ympärillä sen tarjoaman laskentakapasiteetin ja mahdollisuuksien vuoksi. Pilviympäristössä käytössä ovat

käytännössä kaikki mahdolliset työkalut, ja malleja on mahdollista opettaa ja validoida erittäin suurilla määrillä laskentatehoa. Pilvilaskenta tekee suurien datasettien kanssa toimimisesta helppoa. Lisäksi datan saatavuus yleensä paranee, kun se tallennetaan organisaation yhteiseen tietovarastoon, parhaassa tapauksessa tällä voidaan estää organisaation siiloutumista.

Question 8: What is the state of the art for data storage?

Myös tässä teemassa State-of-the-artin määrittely on hankalaa. Tähän mennessä on tieto usein varastoitu MPP-kolumnaarisiin kantoihin, mutta uusissa ratkaisuissa päädytään yleensä jonkinlaisin data lake -ratkaisuihin. Data lake -ratkaisuissa tieto tallennetaan levylle (esim. AWS:n S3), mutta siihen voidaan kohdistaa SQL-kyselyitä.

Toisaalta EMR ja Hadoop ovat kasvattamassa suosiotaan, joissa myöskin tieto tallennetaan levylle SQLtietokannan sijaan. Monesti tästä tiedosta kaivettu informaatio voidaan puolestaan tallentaa tietokantaan esim. visualisointia varten. Lisäksi uusin kuuma trendi tietovarastoinnissa on Snowflake, joka separoi tiedon tallennuksen ja laskennan. Siinä tieto tallennetaan AWS:n S3:een, jonka päällä pyörii skaalautuva määrä EC2-virtuaalikoneita. Sekä S3 tallennustilaa ja EC2 määrä voidaan skaalata toisistaan riippumattomasti, kun taas perinteisissä tietovarastoissa laskentateho ja tallennustila ovat sidottu toisiinsa.

Tämän voisi oikeastaan tiivistää siten, että Snowflake on tällä hetkellä erittäin suosittu ja hyväksi havaittu teknologia, mutta myös hyvin rakennettu data lake on myös hyvä ratkaisu.

Question 9: How complicated are data exits if a cloud platform change is required?

Jos tieto on tallennettu data lake-muodossa, siirtämisessä ei ole mitään ihmeellisiä ongelmia, kaikki palvelut tukevat tällaista dataa. Tietysti datan siirtämisessä kestää, varsinkin mikäli sitä on kertynyt suuria määriä. Tällaisissa tilanteissa datan osittainen siivous voi olla myös järkevää.

Mikäli analytiikassa on käytetty avoimen lähdekoodin kirjastoja, ei mitään erityisiä ongelmia pitäisi ilmetä analytiikankaan siirrossa. Koska analytiikka on enemmän ja enemmän ohjelmakoodia, eikä koodia oikeastaan kiinnosta missä sitä suorittava virtuaalikone sijaitsee, ovat migraatiot usein kohtalaisen suoraviivaisia. Mikäli toteutettavasta sovelluksesta haluaa täysin alustariippumattoman, sen voi toteuttaa esim. Docker-konteilla.

Pilvipalvelut paketoivat monia toiminnallisuuksia omiksi palveluikseen, kuten esim. jonokäsittelyn. Näissä toteutuksissa jokaisella on omat nyanssinsa, mutta pääpiirteittäiset toimintaperiaatteet ovat kaikilla oikeastaan samoja. Tällöin toiminnallisuuksia voi joutua rakentamaan osittain uusiksi migraatiossa. Näiden alustaspesifien palveluiden hyödyt (helppous, kehityksen nopeus) ovat kuitenkin monesti haittoja paljon suurempia, jonka vuoksi ei ole usein järkevää tehdä täysin alustariippumattomia toteutuksia.

Migraatioista puhuessa on hyvä myös miettiä miksi migraatio halutaan ylipäätään tehdä. Monissa tilanteissa parempi ratkaisu saattaa olla ns. multicloud-malli, jossa eri kompo-

nentit sijaitsevat eri pilvipalveluissa. Tämä malli voi sopia esim. tilanteisiin jossa toinen pilvipalvelu tarjoaa jotakin toiminnallisuutta, mitä nykyiseltä tarjoajalta ei löydy. Lisäksi tällä mallilla voi saavuttaa rahallisia etuja, mikäli jokin toiminnallisuus on hinnoiteltu jossain toisessa palvelussa halvemmaksi.

Question 10: How common are multi-cloud solutions in analytics?

Meillä on muutamia monipilviratkaisuja joissa ei ole ollut juurikaan ongelmia. Kaikki nämä ovat kuitenkin kohtalaisen pistemäisiä ratkaisuja ja vaihtoehtoisessa pilvipalvelussa ajetaan vain muutamaa komponenttia. Monipilviratkaisut sopivat erittäin hyvin sovelluksille, jotka on toteutettu mikropalveluarkkitehtuurilla, koska siinä yksittäiset komponentit ovat linkittyneet toisiinsa löysästi ja eivät vaadi, että niitä ajetaan samalla alustalla.

Question 11: From data analytics' point of view, what is the difference between stream data and batch data?

Yleensä vaatimukset sisään tulevan datan muodosta (eräajo vs. jatkuva datavirta) pitää johtaa käyttötarkoituksesta. Mikäli dataa analysoidaan algoritmilla, joka ajetaan kerran yössä, ei kannata juurikaan käyttää resursseja reaaliaikaisen datavirran luomiseksi. Tietysti mikäli tällainen on jo valmiiksi käytössä, kannattaa se hyödyntää sellaisenaan, koska se mahdollistaa tulevaisuudessa reaaliaikaisemman analytiikan.

Eräällä asiakkaallamme on käytössä analytiikka, joka kertoo käyttäjälle välittömästi, mikäli prosessissa tapahtuu poikkeama. Tällaisessa käyttötapauksessa reaaliaikainen datavirta on ehdoton vaatimus. Yleisesti voisi sanoa, että järjestelmille asetettavat vaatimukset ovat kasvamassa, ja entistä useammassa käyttötapauksessa tarvitaan reaaliaikaisia datavirtoja eräajojen sijaan.

APPENDIX B: INTERVIEW WITH CUSTOMER'S AUTOMATION EXPERT

Question 1: What kind of automation logics does the line in question have.

Tarkasteltavassa linjastossa kaikki logiikat ovat Siemensin S7-logiikoita. Suurin osa on 2000-luvulta, mutta vanhimmat ovat vuodelta 1998. Kaikilla logiikoilla ei ole Ethernet-kortteja, mutta niihin on mahdollista lisätä sellainen jälkikäteen. Laitteita päivitetään pääsiassa vain, mikäli siihen liittyy jonkinlainen business-case.

Question 2: Is OPC or OPC-UA used currently?

Meillä ei ole käytössä prosessin ohjauksissa OPC:ta tai OPC-UA:ta mihinkään, koska valvomomme toimivat Siemensin valvontaohjelmistoilla (PCS7 ja WinCC).

Question 3: How does the current data collection look like?

Paljon prosessille relevanttia data tallennetaan valvomokoneille .txt-tiedostoihin jo tällä hetkellä, mutta valvomossa ei ole ollut tapoja/tarvetta esittää tätä kaikkea dataa. Monia arvoja prosesseista tietoa on kerätty jo kauan, mutta mielenkiinto sitä kohtaan on herännyt vasta viime aikoina, erityisesti koska prosessissa on havaittu materiaalihävikkejä. Esimerkiksi punnitustietoja on kerätty talteen jo kauan, mutta tapa siirtää ne relevanttiin paikkaan ja tiedon esitys on puuttunut. Lisäksi joitakin arvoja mitä logiikoissa on ei tallenneta, koska kyseisen datan keräys ja esitys ei ole ollut projektien skoopeissa. Juuri tämän ongelman ympärille on kokeiltu Inductive Automationin Ignition-tuotetta, jolla on pyritty lisäämään näkyvyyttä toteutuneisiin punnitukseen, jotta niitä voidaan vertailla suunniteltuihin arvoihin. Tämä kokeilu on rajoittunut yhteen linjaston prosessikokonaisuuten. Valvoimoissa mittausdata on tallennettu Siemensin valvomoohjelmistojen tietokantoihin, siellä data on tallennettuna 2-4 viikkoa, jonka jälkeen vanhin data poistuu (FIFO).

Question 4: Roughly how much relevant data is generated per hour?

Tällä hetkellä kuukauden aikana tallennettu data vie alle 2 GB tilaa.

Question 5: How does the factory networks look like? How hard is it to get Internet connectivity?

Tehtaan lattialla on kaksi eri verkkoa: toimistoverkko ja tuotantoverkko. Nämä verkot on eriytetty virtuaaliverkoilla (VLAN). Tällä hetkellä kehityssuunta on kohti palomuurilla erotettu verkkoja.

APPENDIX C: INTERVIEW WITH CUSTOMER'S HEAD OF DIGITAL TRANSFORMATION

Question 1: Does your company have a cloud or a data strategy

Meillä on juuri valmistunut konsernin laajuinen (tekninen) datastrategiaehdoitus, jossa tähdätään nykyisten perustietovarastojen siirtämiseen täysin pilveen jollain aikavälillä. Liitteenä on ehdotelmia arkkitehtuurista ja ehdotelma käytettävistä palveluista Azuren pilvessä. Tämä datastrategia tehtiin hyvin liiketoimintalähtöisesti ja sen pohjana käytettiin liiketoiminnoissa havaittuja tarpeita.

Question 2: For what purposes do you want to add analytics capabilities to your organization?

Taustalla on monia hyvin globaaleja trendejä ja ongelmia, meille relevantti ongelma on globaali ruokahävikki – jopa 30% tuotetusta ruuasta päätyy jätteeksi. Nykymaailmassa yritetään myydä mahdollisimman paljon riippumatta siitä mitä vähittäiskauppa tai asiakas oikeasti tarvitsee. Me haluamme tietää asiakkaamme tarpeen ja myydä heille juuri oikean määrän juuri oikeaan aikaan. Tietysti myös omissa tuotantoprosesseissamme on kehitettävää, esim. hävikin pienentämisen ja tuotannon optimoimisen saralla.

Question 2: What is the goal for data gathered from the production line where this thesis' scope is limited? Is it to optimize the line in question or is the end goal to provide corporation level enablers (like data fusion including production data)?

Primäärinen tarkoitus on hankkia näkyvyyttä tuotantoprosessiin, hankkia peruskyvykkyyksiä ja lisätä maturiteettiä kohti Industriy 4.0 :sta. Tällä hetkellä integraatioita on vähän liiketoiminnan ja tuotannon välillä, ja tieto ei synkronoidu näiden järjestelmien välillä automaattisesti vaan hyvin moni asia on ihmisten varassa. Jatkossa tavoitteena on ymmärtää kaikkea, esimerkiksi tehdä skenaarioanalyysejä, ymmärtää riippuvuuksia ja havaita esim. vaikuttaako vuoroprofiilit hävikkeihin yms. Haluamme myös tehdä työntekijöidemme työstä merkityksellisempää ja toteuttaa uusia tehokkuus- ja laatumittareita. Näiden avulla työntekijät voivat oikeasti ymmärtää miten he voivat toimia tehokkaammin ja nähdä oman toimintansa tuloksen. Tällä hetkellä yksilölle on hyvin vaikeaa ellei mahdotonta nähdä miten hänen työpanoksensa vaikuttaa koko tehtaan tuotantoon.

Question 3: What kind of an architecture have you envisioned regarding data analytics for production data

Olemme tällä hetkellä tutkimassa tarvetta MOM / MES -järjetelmälle ja miettimässä sen roolia osana tulevaisuutta. Henkilökohtainen mielipiteeni on, että tarvitsemme jonkinaisen

MOM-järjestelmän/järjestelmiä luomaan prosessiautomaation dataan kontekstia. Lisäksi modernin MOM-järjestelmän pitäisi integroitua esim. pilvialustoihin tai suoraan SAP:iin helposti. Voi tietysti olla, että projektimme lopputuloksena ymmärrämme, että emme tarvitsekkaan tällaista perinteistä MOM-kerrosta, vaan sen korvaa esim. Kepserver. Mikäli kuitenkin MES nähdään tarpeelliseksi osaksi infrastruktuuria, sen suhde edistyneeseen analytiikkaan on mielenkintoinen. Hoitaako se ns. data engineeringin analytiikkaa varten, välittääkö sen kenties datan hyvin minimaalisilla muutoksilla jollekkin analytiikka-alustalle vai hoitaako jonkinlainen MES:n moduuli miltei kaiken analytiikan? Lisäksi yhtenä kysymyksenä on se, onko SAP HANA meille väistämättä tulevaisuus koska olemma vahva SAP-talo.

Question 4: Why do you think your company needs a MES (in the factory/production line in question)?

Tärkeimpänä syynä pidän kontekstitiedon tarjoamista prosessiautomaation datalle. Toisena tärkeänä tehtävänä pidän tehdaslattian orkestrointia. Kolmantena tärkeänä tehtävänä pidän ei-konventionaalisen datan hallinta, esim. tiettyyn tuotantoerään tms. liittyvät kuvat.

Question 4: Do you have preferences regarding open source vs. products

Meillä ei ole mitään virallista strategiaa liittyen avoimeen lähdekoodiin suhteessa tuotteisiin. Ongelma on pääasiassa ollut tapamme hankkia eri puolella organisaatiota hyvin pistemäisiä ratkaisuja yksittäisiin ongelmiin ja näiden harmonisoinnissa on suuri työ.

Question 5: Analytics timeframe, are we searching for optimization on a second and minute basis or mainly for longer basis like days and weeks?

Jossain vaiheessa haemme sekuntitason optimointia tuotantoon.

Question 4: What kind of a life span is planned or required for analytics platform

Pilvipalvelut ovat vain palveluita. Tietysti isojen datamassojen siirrossa on omia ongelmiansa, mutta uskon vahvasti multi cloud -strategiaan.